Underwater Video Analysis for Fish Biodiversity Monitoring

Wang Yangxuan
Department of Computer Science
University of Exeter
Exeter, UK
yw648@exeter.ac.uk

***Abstract*—Marine biological monitoring and protection have always been a hot spot in the scientific community. Many advanced computer vision technologies, such as moving object detection, object trajectory tracking, semantic image segmentation, Etc., have been widely used in the underwater environment analysis field, providing a reliable tool for marine biologists. We have built a fish detection framework that combines image enhancement algorithms and pre-trained CNN-based models, providing marine biologists with automated tools. This model can monitor and analyze realistic underwater video fish populations collected by the BRUVs method. The water environment posed many challenges to visual tasks, so we tried traditional image enhancement methods and Retinex-based image enhancement algorithms, then perform migration training on YOLOv5. Test results of two public real underwater video datasets are mean average precision (mAP 0.5) is 99.2% and 98.7% respectively, both scores exceeded the baseline provided by dataset official.**

***Index Terms*—Object detection; Data augmentation; Retinex; Fish identify; Deep learning; Transfer learning; YOLOv5; CNN; Computer vision**

I. Introduction

The ocean is the main component of the earth's surface. In addition to food, medicine, and materials, the vast ocean also provides leisure, tourism, and research resources to humans; all kinds of marine life made a significant contribution to the oxygen circle of the earth, climate regulating and decompose waste that generates by other creatures [1]. Fish biodiversity monitoring is an essential part of marine research. Fish abundance, species distribution, sex ratio, behavioral characteristics, Etc., are basic information of marine analysis [2]. Scientists can estimate the climate change of a specific area according to the changes in population abundance of a specific certain marine life [3]. Otherwise, fish classification helps track their motion trajectory and migration trend and building a deeper understanding of species [4]. However, relying on oceanographers to manually process and analyze these video data is relatively time-consuming, subjective, and labor-intensive [5], so the research of automated underwater video fish detection and analysis has a good development prospect [6]. Generally, such research based on massive underwater data, the robustness, accuracy, and unbiasedness of dataset is fundamental [2]. Baited Remote Underwater Video (BRUV) is an underwater video collect technology that is rapidly growing in marine ecology due to its accessible, affordable, low biases, and non-invasive advantages [7], [8]. The underwater environment usually has posed challenges of low resolution, insufficient light, complex background, occlusion and deformation caused by object movement, the similarity between different fish species, etc. [9], [10]. In addition to change model architecture and use better algorithms, data augmentation can also improve the performance of object detection models in underwater tasks. Meanwhile, it avoids making the model too complex to lower inference speed [11], and they are well documented to be useful in supplement datasets and avoid overfitting [12]. Otherwise, deep learning-based methods have become the recent trends of fish identification, conventional neural networks (CNN) and their variants are acknowledged to be state-of-the-art in image classification research [6].

This project aims to apply computer vision and deep learning technology to the datasets collected in the Baited Remote Underwater Video (BRUV) surveys for non-invasive fish biodiversity monitoring. In response to water environment challenges, built an efficient work pipeline, identify fish that appear in videos, and classify their species. Our framework combines data augmentation algorithm with transfer learning on pre-trained CNN model YOLOv5 [13], explored the performance of traditional augment methods and Retinex-based method in such underwater tasks. We trained and tested our framework on two different public underwater fish datasets, LifeCLEF-2015 [14] and Brackish Underwater [15] and achieved satisfactory results.

The rest of this report is organized as follows: the second part is related work, which discussed some specific research results of underwater fish species identification; the third part is aims and objectives, the goal of the experiment will be clearly described, also the expected results; the fourth part is experiment design, which consists of dataset description, algorithm principle, criterion, benchmark, and experiment strategies; all results will be presented, compared and discussed in fifth part; the last part is the conclusion and future works.

II. Background & Related work

This part presented the background of the main technologies used in this project and some reference research and their results. Part A is methods of underwater video collection, part B is about techniques of underwater data augmentation, while

object detection includes fish identification, will be discussed in part C.

A. Underwater data collection

Traditional underwater data collection techniques include Underwater Visual Census (UVC), Diver Operated Video (DOV), Automated Underwater Vehicles (AUV), baited traps, angling, trawling, netting, ship-based sonar, etc [16]. However, there are some limitations of these destructive methodologies that have been confirmed [8]. The movement of the divers and underwater robotics will change the behavior of the fish, thereby biasing the survey data [17]; also, the water areas that divers and robots can investigate are limited by water depth and visibility, which also filters the fish species to a certain extent [18]. Traps and net fishing methods not only easy to cause harm to animals and habitat, the type and size(mesh) of tools also result in extractive and size-selective data [19]. Acoustic-based methods can effectively detect the fish population abundance and biomass but cannot provides accurate species information [20].

Consequently, Baited Remote Underwater Video Stations (BRUVS) becomes a popular sampling methodology, particularly on fish assemblage studies [16]. This technology use bait to attract fish into the field of view of a remote-controlled camera, avoid the impact of direct contact between divers and fish. Many research compares BRUVS with traditional methods [19], [21]–[23], and conclude BRUVS has advantages of non-invasive, non-extractive, cheaper personal cost, safer and easier operation [16]. Moreover, in long-time survey tasks, BRUV is cost-effective, and it can investigate some areas that human divers are hard to access [7], [24]. Through these underwater videos collected by BRUV, researchers can have the opportunity to observe the behavior of fish in the natural environment to obtain more reliable results [25].

Based on BRUV data, Brooks et al. analyzed the diversity, distribution, and abundance of Bahamas sharks, they conclude that BRUV is a viable alternative to the longline survey, but due to the changeful activity of the fish in the front of the collect equipment, BRUVS has no outstanding performance in distinguishing fish species, size, and sex, researchers should choose a relatively appropriate data collection method according to the purpose of the experiment [19] Another relative abundance study is conducted by Jabado et al. in Arabian Gulf. Their BRUV survey is collected 278 underwater videos, which have 757 hours of soak time, recorded 213 instances of sharks and rays in 20 species. This research proved that the BRUV sampling method could be used in the analysis of elasmobranchs across broad spatial scales, especially in areas where difficult to apply traditional fisheries techniques [26]. At the same time, the study of Taylor et al. indicates that when the BRUV technique is conducted in estuaries and coastal waters, it is essential to consider impacts of tidal and currents/citetaylor2013tidal .

B. Data augmentation

Most of the object detection works focus on constructing a better model framework or more intelligent algorithms. At the same time, it leads to more accurate models and makes the model more complex, significantly reducing the detection speed. Data augmentation, however, also boosts detection performance by improving the data quality but not the architecture of models, therefore keep the efficiency [11].

Moreover, the training of deep convolutional neural networks required a great amount of data, but not all application domains have access to big data, such as underwater video and medical image [27]. To solve the overfitting problem, many strategies have been researched, such as Dropout [28], Batch normalization [29], pretraining [30], etc. Different from these methods, the data augmentation technique expands the dataset, avoid overfitting from the root [27].

Standard image enhancement methods include geometric transformations (flipping, rotation, shear, scale, Etc.), color space augmentations (hue, saturation, value, Etc.), kernel filters, mixing images, random erasing, feature space augmentation, adversarial training, generative adversarial networks, neural style transfer, and meta-learning, etc [27]. Besides, Retinex is an image enhancement technique that is often used to deal with insufficient light in the image [31], in recent years, some image processing techniques based on the Retinex theory have also been used in underwater image tasks [32]–[34]. Most of the changes in the water environment are due to the changes in light and the scattering in water evenly distributed, which makes the application of Retinex in underwater pictures possible [32].

NgoGia et al. used Mosaic-based data augmentation on sea cucumber detection. Before Mosaic, they applied geometric augmentation and color space augmentation to images then shuffle them. Compare with before augment, the mAP (IoU threshold = 0.5) has been increased by 24.47% [35]. In the study of Huang et al., three different data augmentation methods are proposed, the inverse process of underwater image restoration, perspective transformation, and illumination synthesis. In the task of detecting and recognize marine organisms, their proposed techniques improve the robustness of Faster R-CNN on aspects of variable turbulence, variable shooting angle, and uneven light underwater, respectively [36]. Fu et al. proposed a Retinex-based augment method used in the single underwater image, which effectively solves the blur and underexposure problems of a single underwater image, correct image color, and preserves well sharpness and brightness, and naturalness. This method also can be used in other images that need enhancement [32]. In a three-stage underwater image augment approach proposed by Hassan et al., Retinex is used in underwater images that preprocess by CLAHE technique, restore the colors by divide color into reflectance and luminance [37].

C. Object detection and classification

In object detection filed, early research use Histogram of Oriented Gradients (HOG) [38], Speeded Up Robust Features

(SURF) [39], Local Binary Pattern (LBP) [40], and principal component analysis (PCA) [6] to extract features, however, past decade, machine learning based technical especially deep learning are popular and shown powerful performance. Conventional neural network (CNN) is the most classic deep learning model, and it has amazing capability in recognizing complex structures in high dimensional data [41], which has advantages of hierarchical feature representation, deeper architecture, jointly optimize and large learning capacity, consequently, it has been widely used in image classification, human face recognition, pedestrian detection, and video analysis, etc. filed [42]. CNN-based object detector includes two-stage (RCNN [43], SPPNet [44], Fast-RCNN [45], Faster-RCNN [46], etc.) detector and one stage detector (YOLO [47], SSD [48], Retina-Net [49], etc.), the former generates candidate box first, while the latter complete all tasks in "one step".

Since building a new CNN model requires many training data, and the underwater fish video data is relatively insufficient, the use and pre-trained model for transfer learning has also become a popular method in such tasks [9]. Transfer learning means training network on a big dataset like ImageNet, and then save this weight as the initial weight of another specific task [27]. Various pre-trained CNN variant such AlexNet [50], Fast-RCNN [45], VGG [51] and YOLO [47]etc. have also been used in fish species identification.

Jalal et al. proposed an underwater video classification method that combines deep learning and time information [10] This method combines the optical flow model and Gaussian mixture model with the YOLO deep neural network can detect freely moving fish in a complex environment, effectively improving the performance of the classification model in underwater videos. On the data set provided by the University of Western Australia (UWA), this systematic fish monitoring achieved a score of 91.2%. The correct rate of fish species discrimination reached 79.8%, which is a considerable result and provides a practical reference for us to solve the light intensity, the turbidity of the water, the dynamics of the fish, and other challenges [10]. Yuan et al. combined Multi-Scale Retinex with Color Restoration (MSRCR) with Faster R-CNN, then transfer learning on a high-resolution fish image dataset. This experiment trained an effective fish detection model on a small dataset, and its performance exceeds traditional methods without data augmentation/citeyuan2020underwater.

On the other hand, Qin et al. proposed a deep architecture fish recognition framework that uses the deep architecture to extract fish features, extracts information from the spatial pyramid pool (SPP), and then uses linear support vector machines (SVM) for classification. This framework uses video foreground extraction based on sparse and low-rank matrix decomposition, effectively reducing noise and can identify fish even in complex background environments. In the Fish4Knowledge (F4K) repository, the accuracy of DeepFishSVM framework fish recognition is higher than 98% [52]. Siddiqui et al. proposed a new computer vision alternative model that uses a pre-trained deep convolutional neural network combined with cross-layer pooling technology to detect image features and then use the Support Vector Machine (SVM) to classify the calculated features. This method dramatically reduces the need for training data and avoids over-fitting problems of the model, In the data set LifeClef'15 from the Fish4Knowledge (F4K) repository, an accuracy rate of 96.73% is reached, and it is also obtained good results in an insufficient training data [6].

III. Aims & Objectives

Fish biodiversity monitoring includes many different parts, such as fish tracking, population abundance estimating, fish size analyzing, Etc. This project focuses on classifying species, automatically identifying fish in underwater videos, and identifying their specific species.

Our fish identify frame based on deep learning, which has achieved remarkable improvement in multi-classification visual detection problem [53], and there are research finds that it has surpassed human in species classification task [2]. The pre-trained ConvNet we chose is YOLOv5, which is a one-stage detector, provides different architecture. We performed transfer learning on LifeCLEF-2015 [14] and Brackish Underwater [15], which are two public underwater fish datasets provided by Fish4Kownledge repository [54] and Aalborg University respectively. Although the two data sets are underwater fish videos, their collection sea areas, challenges, fish species, the number of classes, and the classification indicators are quite different.

Another critical part of this project is data augmentation. We applied geometric transformations, HSV color space augmentation, autoMSRCR (automated Multi-Scale Retinex with Color), and data balance on two datasets and then compare the classify results, analysis performances of data augmentation techniques on different underwater challenges. Both datasets provide official fish classification baseline scores for researchers to contrast, and our proposed framework aims to exceed the score.

IV. Experiment Design & Methods

In the beginning, the overview of the experiment pipeline and strategies will be presented in part A. Part B will detail describe two datasets separately. All data augmentation and data process operation methods also are introduced in part C. Part D is about the pre-trained model, the structure of YOLOv5, and the reasons we chose it. In the last part, we discuss the evaluation criteria of classification performance and the experiment benchmark.

A. Overview of Experiment

The flowchart is shown in Fig 1 clearly described all steps of this experiment. The first step is decomposing all videos into single pictures. To avoid miss fish information, we extracted all frames, and each frame has its corresponding annotation. All frames and all annotations are stored separately in two folders. The second step is transforming the original annotation to the YOLOv5 model required format and then applied data cleaning on the LifeCLEF-2015 dataset. The details will be described in part C of this section.

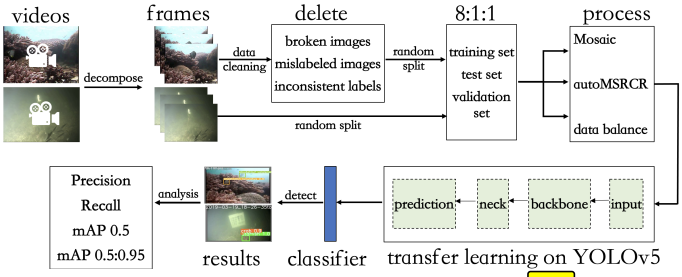


Fig. 1: Main structure of this proje

The next step is to randomly split all pictures into the training set, test set, and validation set according to the ratio of 8:1:1.

The fourth step is data augment processing, and based on previous preparation, is use YOLOv5 model for transfer learning on two different datasets separately. It is worth mentioning that based on different data augment methods; we designed five different training strategies:

- *Strategy 1*: Origin data
- *Strategy 2*: Only applied Mosaic augmentation on the dataset
- *Strategy 3*: Only applied autoMSRCR on the dataset
- *Strategy 4*: Applied Mosaic augmentation, and autoMSRCR on the dataset
- *Strategy 5*: Applied Mosaic augmentation, autoMSRCR, and data balance on the dataset

Correspondingly, we got five fish detectors trained by different strategies, then use them to detect fish and classify species of the test set. Finally, analysis all results and conclusion.

B. Dataset & Resources

1) LifeCLEF-2015:

The first underwater dataset comes from a video-based fish identification competition hold in 2015 [14]. It contains 93 videos in total, and each video is manually annotated by two experts and save as the XML file, which records all bounding boxes of fish instances and their species. Fig 2 presents sample

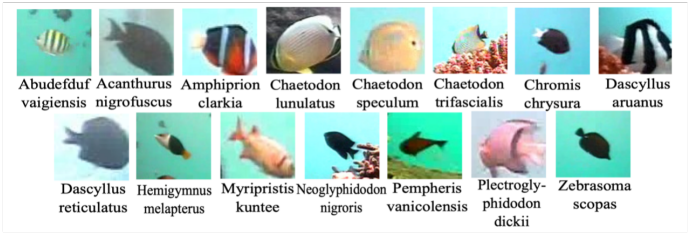


Fig. 2: 15 fish species sample images of LifeCLEF-2015 [9]

images of different species; 15 kinds of fish are recorded in this dataset. Some fish species, like Amphiprion clarkia and Myripristis kuntee, has bright colors and easily distinguish patterns on their skin, while some fish species like Acanthurus nigrofuscus and Dascyllus reticulatus, look black, which can only be distinguished from the outline of the fish. We got 14381 labeled frames after decomposed all videos, and Fig 3 are sample frames extracted from videos. We can easily conclude that this video dataset is collected on the coral reef area with relatively sufficient light and clear seawater. More detailed fish examples are shown in the appendix.

However, this also brings some challenges, complex and changeable background, low resolution, the movement of corals and fish caused distortion, the coral obscures the fish body; also many fish in the distance are small black blobs, which is difficult to distinguish fish species even with human eyes. On the other hand, after data cleaning and resplit, Table I

Fig. 3: LifeCLEF-2015 Sample frames indicate challenges like complex background, low resolution, distortion, obscures, Etc.

indicates the instance number of different species; this dataset is imbalanced. There are 12607 fish recorded; the Dascyllus reticulatus is the majority species, with a total of 3369 instances, nearly 20 times the minority species Zebrasoma scopas.

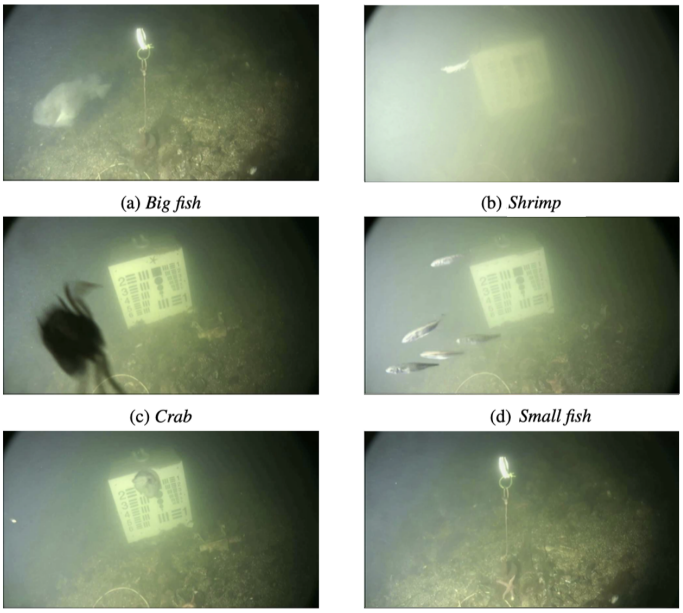
TABLE I: LifeCLEF-2015 Instance number of different species

Species	Training set	Test set	Validation set	Total
chaetodon trifascialis	995	120	109	1224
dascyllus reticulatus	2699	341	329	3369
acanthurus nigrofuscus	302	25	40	367
chaetodon lunulatus	1217	202	231	1650
chaetodon speculum	202	15	11	228
pempheris vanicolensis	1038	45	36	1119
abudefduf vaigiensis	201	19	16	236
chromis chrysura	306	28	29	363
plectrogly-phidodon dickii	763	110	105	978
hemigymnus melapterus	265	20	14	299
myripristis kuntee	274	15	21	310
zebrasoma scopas	129	20	22	171
amphiprion clarkii	395	88	80	563
neoglyphidodon nigroris	147	158	153	458
dascyllus aruanus	894	194	184	1272
Total	9827	1400	1380	12607

2) Brackish Underwater:

This dataset is provided by Aalborg University and recorded in a strait, Limfjorden [15]. It contains 89 videos and annotation files, which have COCO, YOLO, and AUU format. Due to the low visibility, the fish are not subclassified into specific species but roughly divided into six big categories, crab, big fish, small fish, starfish, shrimp, and jellyfish. We extracted

12447 fish recorded frames from all videos; Fig 4 indicates sample frames of all six classes. The main challenges of the brackish dataset are low and variation water turbidity and in-class variation.More sample frames are shown in the appendix. According to the dataset paper, the water deep of the recording



(a) *Big fish* (b) *Shrimp* (c) *Crab* (d) *Small fish* (e) *Jellyfish* (f) *Starfish*

Fig. 4: Brackish sample frames present different fish classes and challenges of low light and turbid water. [15]

location is nearly 15 m deep, and the water is mixed from seawater and freshwater from streams [15]. Currents, winds, and the weather will lead to different visibility of brackish water, while subgraph (a) and (b) of Fig 4 have shown high turbidity and low turbidity, respectively.

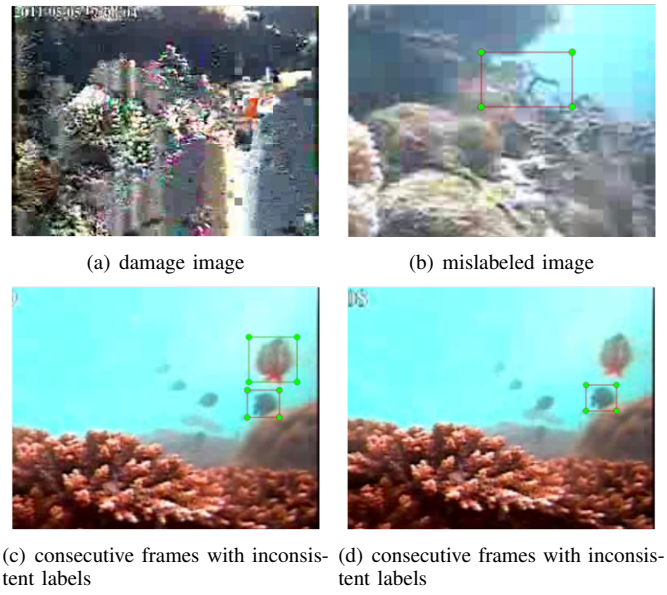
Similarly, this dataset is imbalanced too. Table II shows the distribution of labeled fish instances after random split into training, test, and validation sets. In total, 35565 annotated fish, the most class is crab, which has 12348 instances, and the minor class is shrimp, 548 instances, while jellyfish also only 637. This data set is more unbalanced than the previous LifeCLFE-2015, and the number of majority classes is about 22.5 times the number of minority classes.

TABLE II: Brackish instances distribution

Species	Training set	Test set	Validation set	Total
big fish	2709	321	322	3352
crab	9880	1279	1189	12348
jellyfish	520	62	55	637
shrimp	415	57	76	548
small fish	8655	965	1148	10768
starfish	6339	782	791	7912
Total	28518	3466	3581	35565

3) Resources:

This experiment will be written in python3 language, the YOLOv5 is open source, and we trained it on Google Colab,



(a) damage image (b) mislabeled image (c) consecutive frames with inconsistent labels (d) consecutive frames with inconsistent labels

Fig. 5: Bad data samples of LifeCLEF-2015

based on Pytorch 1.9.0, Cuda 10.2. Colab pro usually provided GPU like tesla P100 or V100. Because of the limited GPU resource, users cannot determine the GPU model. Also, the use time is limited, not enough for complete training of the model. As a result, the training process is carried out in stages and the type of GPU automated allocated each time by Colab is different.

For the above reasons, this project only evaluates the model's classification accuracy, the comparison of the convergence rate of the model is rough, and the inference speed is temporarily not considered.

C. Data process

1) Data cleaning:

Only the first dataset, LifeCLEF-2015, is cleaned. Since the Fish4Knowledge official website has ceased maintenance, our LifeCLEF-2015 dataset comes from a private GitHub repository. There are too much bad data in it, and many papers that use this dataset mentioned they amend some miss labeled data. As a result, we consider it is necessary and has sufficient reason to implement data cleaning. The data cleaning mainly includes manually checking fish frames extracted from videos, deleting damaged images, mislabeling images, and consecutive frames with inconsistent labels. Fig 5 presents examples of frames that need delete, subfigure (a) is a damaged image, which may be caused by compression and decoding after video capture. There is no fish in subfigure (b), but the bounding box of the fish is marked. Subfigure (c) and (d) are two consecutive frames extracted from a video, and the fish targets contained in them are the same. However, two fish are marked in (c), and only one is marked in (b), so we delete the missing frames. Due to the massive amount of data, we only deleted a part of such bad data. While cleaning the data, it maintains

the characteristics of the naturally collected data, which is conducive to enhancing the robustness of the model.

2) Data augmentation:

Data augmentation methods that used in this experiment are Mosaic and autoMSRCR (automated Multi-Scale Retinex with Color Restoration)

- ***Mosaic***: Mosaic technique based on geometric transformations and HSV color space augmentation uses four training images to stitch together, dramatically increases the complexity of the background, and has a good effect on the detection of small targets. Specifically, first, flip, scale, rotate, and adjust hue, saturation, and value of four selected pictures, then place processed pictures according to the first picture on the upper left, the second picture on the lower left, the third picture on the lower right, and the fourth picture on the upper right. After placement, capture the specific areas of the four pictures and then stitch them together to form a new picture, which contains object boxes. Fig 6 shows four original pictures and the Mosaic result picture. The new picture contains more information than before, but not all fish objects are displayed because the pictures block each other.

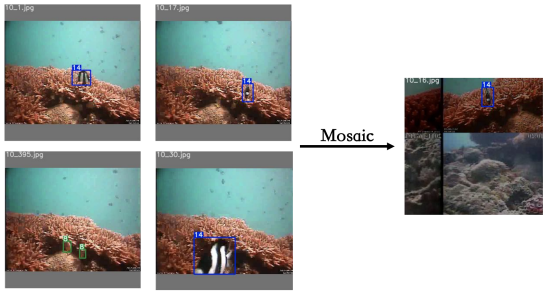


Fig. 6: Mosaic augmentation sample of LifeCLEF-2015

- ***autoMSRCR***: Image processing algorithms based on Retinex theory include SSR (Single Scale Retinex), MSR, MSRCR, and autoMSRCR. Among them, MSRCR and autoMSRCR improve the color cast problem of the picture by introducing color restoration operators; both are significant improvements of the Retinex based method.
  SSRis the basis of other methods. According to Retinex theory, an image is composed of incident light and reflected light:

$$T(x,y) = R(x,y) \cdot I(x,y) \tag{1}$$

where T(x, y) is the true image, R(x, y) is the reflect image, which represents the intrinsic reflection property of the object, and I(x, y) represents the incident image. The observed image o(x, y) can be express like:

$$o(x,y) = logR(x,y) = log\frac{T(x,y)}{I(x,y)} \tag{2}$$

$$o(x,y) = logT(x,y) - log[G(x,y) \otimes T(x,y)] \tag{3}$$

where $\otimes$ in the formula 3 is convolution operation and $G(x,y)$ represents Gaussian surround function, which can be calculated by:

$$G(x,y) = \lambda e^{\frac{-(x^2+y^2)}{s^2}} \tag{4}$$

where s is the Gaussian surround scale and λ is a scale of normalization, and their values must satisfy the following relationship:

$$\iint G(x,y)dxdy = 1 \tag{5}$$

Based on SSR, the expression of MSR is as follows:

$$o(x,y) = \sum_{n}^{N} w_k\{logT(x,y) - log[G_k(x,y) \cdot T(x,y)]\} \tag{6}$$

where N is the number of Gaussian scales, when $N = 1$, MSR becomes SSR. Generally, the value of N is 3, and there are:

$$w_1 = w_2 = w_3 = \frac{1}{3} \tag{7}$$

MSRCR add the color balance, normalization, and linear weighting of gain and deviation on MSR results, the formula is:

$$R_{MSRCRi}(x,y) = C_i(x,y)R_{MSR_i}(x,y) \tag{8}$$

$$C_i(x,y) = f[I_i^{'}(x,y)] = f[\frac{I_i(x,y)}{\sum_{j=1}^{N} I_j(x,y)}] \tag{9}$$

$$\begin{aligned} C_i(x,y) &= \beta log[\alpha I_i^{'}(x,y)] \\ &= \beta\{log[\alpha I_i^{'}(x,y)] - log[\sum_{j=1}^{N} I_j(x,y)]\} \end{aligned} \tag{10}$$

Where $l_i(x,y)$ represents the image of the i-th channel, C_i is the color restoration operators of the i-th channel, β represents the gain constant, and the nonlinear strength is control by α.

autoMSRCR is based on the MSRCR. By removing the maximum and minimum values of the absolute color scale, and then mapping the remaining color scale to the 0-255 color channel, automatic color level adjustment is realized. It is the most effective method in the Retinex series [55].

Fig 7 presents the original sample images and autoMSRCR augmented sample images of two datasets. In LifeCLEF-2015, the picture's brightness has been increased, and in the second dataset, the contrast is improved, and the fish is more visible than the original picture.

3) Data balance:

From the data set description part, both datasets used in this experiment have an imbalance problem, and the data tilt will affect the accuracy of the classification model, especially for small classes. Data balance is the third data process method we applied. Specifically, first, we count the number of instances of each class in the training set and then set a threshold. If the

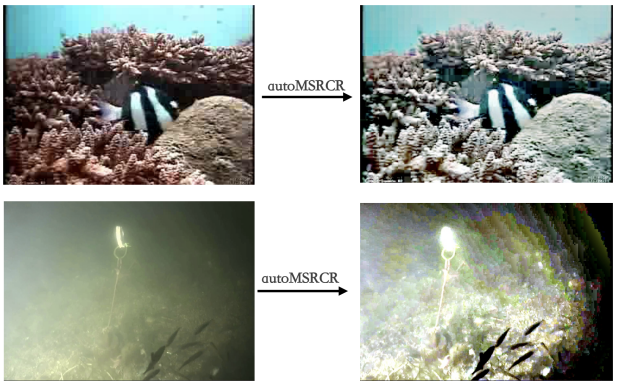


Fig. 7: Sample images of autoMSRCR

number of instances is less than the threshold for each class, we randomly select pictures containing this class instance and use the data augment method (random flip, rotation, adjust HSV) to expand. Table III and Table IV are the statistics of

TABLE III: LifeCLEF-2015 training set instance number before and after data balance

Species	Before balance	After balance
chaetodon trifascialis	995	995
dascyllus reticulatus	2699	2699
acanthurus nigrofuscus	302	606
chaetodon lunulatus	1217	1217
chaetodon speculum	202	415
pempheris vanicolensis	1038	1038
abudefduf vaigiensis	201	537
chromis chrysura	306	651
plectrogly-phidodon dickii	763	813
hemigymnus melapterus	265	400
myripristis kuntee	274	543
zebrasoma scopas	129	530
amphiprion clarkii	395	579
neoglyphidodon nigroris	147	335
dascyllus aruanus	894	530

TABLE IV: Brackish training set instance number before and after data balance

Species	Before balance	After balance
big fish	2709	4998
crab	9880	9880
jellyfish	520	1862
shrimp	415	1504
small fish	8655	8655
starfish	6339	6339

the number of instances of different classes before and after the data balance of the two datasets.

D. YOLOv5

YOLOv5 was developed by Ultralytics company; although the YOLO series official has not yet issued relevant acknowledge, YOLOv5 is regarded as an improved version of YOLOv4, becomes a current hot application of object detection field. Compared with other previous versions of YOLO models, YOLOv5 uses the PyTorch architecture instead of the DarkNet, and according to information provided by Ultralytics, the performance of YOLOv5 is better than the EfficientDet, the detect speed on the COCO dataset is 140 FPS, which is much faster than 50 FPS of YOLOv4 [13]. Moreover, the size of YOLOv5 is only 27 MB, while YOLOv4 with DarkNet architecture is 244 MB. In general, YOLOv5 is a lightweight network with very fast detection speed, and its accuracy is comparable to YOLOv4.

Currently, YOLOv5 has four network models with different depths, YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x. Fig 8 indicates the structure of YOLOv5s, which has the smallest layer depth and feature map size. In contrast, other models are based on it have deeper architecture and wider feature map size. Since the large datasets, we selected the YOLOv5x model, which has a deeper and more comprehensive structure. Take YOLOv5s as an example (Fig8), similar to YOLOv4, the structure of YOLOv5 is divided into four parts:

- ***Input***: Mosaic data enhancement, adaptive anchor box calculation, and adaptive image scaling methods are encapsulated on the data input layer of YOLOv5, which are handy tricks for speeding up inference.
- ***Backbone***: Focus is a unique structure in the backbone network of YOLOv5, and its main operation is image slicing. As shown on the structure, the CBL consists of Conv, BN, and Leaky_relu. Refer to the CSPNet structure; the CSP1_X is convolutional layers Concat with X Res unit modules. The SPP uses maximum pooling methods for multi-scale fusion.
- ***Neck***: Different from YOLOv4, another CSP structure, CSP2_X, is used on the Neck part of YOLOv5. The same part is the FPN+PAN structure.
- ***Prediction***: In output put layer, GIoU Loss (generalized intersection over union loss) is used as the loss function of the bounding box, which is an improved version of IoU (Intersection over Union), and its value range is [-1,1], the closer to 1, the larger the overlap between the prediction box and the label box [55], calculated by:

$$IoU = \frac{|A \cap B|}{|A \cup B|} \tag{11}$$

$$GIoU = IoU - \frac{|C(A \cup B)|}{|C|} \tag{12}$$

where A and B are areas of prediction box and ground truth box, the C is the smallest enclosing convex of A and B.

On the other hand, YOLOv5 used Weighted NMS (Weighted Non-maximum Suppress) to filter the object box.

E. Evaluation & Benchmark

1) Evaluation criteria:

This research evaluates model performance from 4 criteria:

- ***Precision***: Evaluating the accuracy of the prediction of the classifier, which is the number of correctly predicted

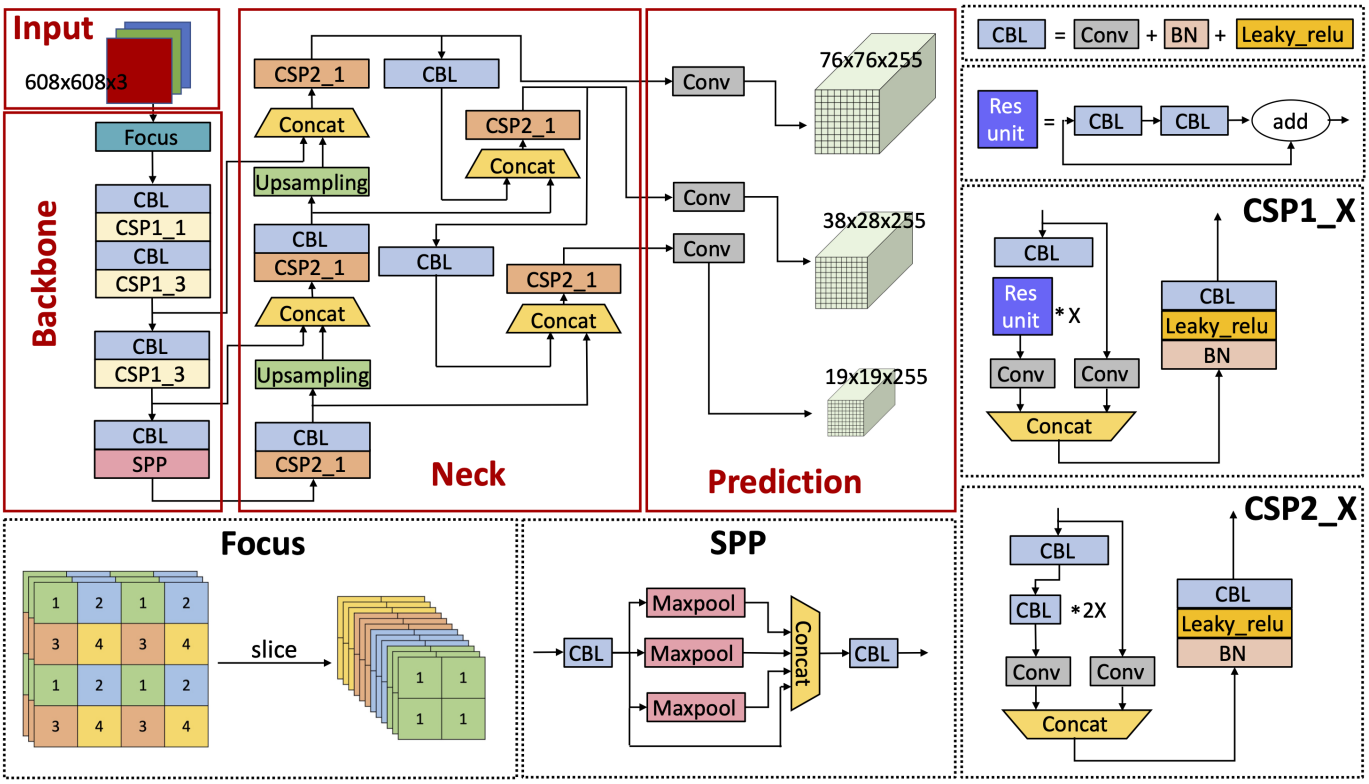


Fig. 8: Structure of YOLOv5s

positive samples divided by the total number of predicted positive samples:

$$Precision = \frac{TP}{TP + FP} \tag{13}$$

- ***Recall***: Evaluating the ability of the classifier to find all positive samples. It is the number of positive samples that are correctly predicted to account for the total number of positive samples:

$$Recall = \frac{TP}{TP + FN} \tag{14}$$

- ***mAP@0.5***: mAP means the mean average precision. AP (average precision) is the area under the PR curve, with can only describe a specific class, and mAP is the average value of AP of all classes, the calculating formula is:

$$AP = \int_0^1 p(r)dr \tag{15}$$

$$mAP = \frac{\sum_{i=1}^{C} AP_i}{C} \tag{16}$$

where C is the number of classes, AP_i is the AP value of i-th class.

The 0.5 is a threshold of IoU (Intersection over Union) when IoU of the predicted box with the ground truth box greater than or equal to 0.5 is considered a true positive sample. mAP@0.5 represents the mAP of IoU=0.5.

- ***mAP@0.5:0.95***: Same as mAP@0.5, 0.5:0.95 means the IoU threshold from 0.5 to 0.95, the stride is 0.05, and mAP@0.5:0.95 is calculated by the average value of all mAP of all thresholds, which can be seen as putting forward higher detection accuracy requirements on the model.

2) Benchmark:

For the LifeCLEF-2015 dataset, the criterion used in this competition is precision, and the best results are recorded on the official website. Top three scores are 0.81, 0.8 and 0.7. In contrast, the Brackish data set provides a more detailed benchmark. This dataset is fine-tuned on YOLOv2 and YOLOv3, and the mAP@0.5 and mAP@0.5:0.95 are 0.0984 and0.3110 on YOLOv2, 0.3893, and 0.8372 on YOLOv5, respectively.

V. Results & Discussion

According to 5 different strategies, we transfer learning YOLOv5x on LifeCLEF-2015 and Brackish datasets, respectively, and got ten fine-tuned models. The detect and classify results of all classes on test sets are indicated below Table V and Table VI: For the first dataset, LifeCLEF-2015, Strategy 5 has the highest mAP@0.5:0.95, 0.835, and the highest precision is 0.985, which is over the top 1 benchmark 0.81. However, for the Brackish dataset, Strategy 4 has the best performance. The mAP@0.5 and mAP@0.5:0.95 are 0.984 and 0.756, respectively, surpass the YOLOv2 benchmark 0.673 and 0.6576, higher than the YOLOv3 benchmark 0.1468 and 0.3667.

TABLE V: LifeCLEF-2015 test result

Training strategy	P	R	mAP@0.5	mAP@0.5:0.95
Strategy 1 (origin)	0.908	0.788	0.85	0.681
Strategy 2 (Mosaic)	0.97	0.98	0.988	0.813
Strategy 3 (autoMSRCR)	0.935	0.795	0.854	0.678
Strategy 4 (Mosaic+autoMSRCR)	0.983	0.975	0.992	0.815
Strategy 5 (Mosaic+autoMSRCR+data balance)	0.985	0.964	0.983	0.835

TABLE VI: Brackish test result

Training strategy	P	R	mAP@0.5	mAP@0.5:0.95
Strategy 1 (origin)	0.618	0.375	0.418	0.201
Strategy 2 (Mosaic)	0.985	0.975	0.987	0.83
Strategy 3 (autoMSRCR)	0.598	0.574	0.575	0.367
Strategy 4 (Mosaic+autoMSRCR)	0.981	0.966	0.984	0.756
Strategy 5 (Mosaic+autoMSRCR+data balance)	0.972	0.952	0.975	0.738

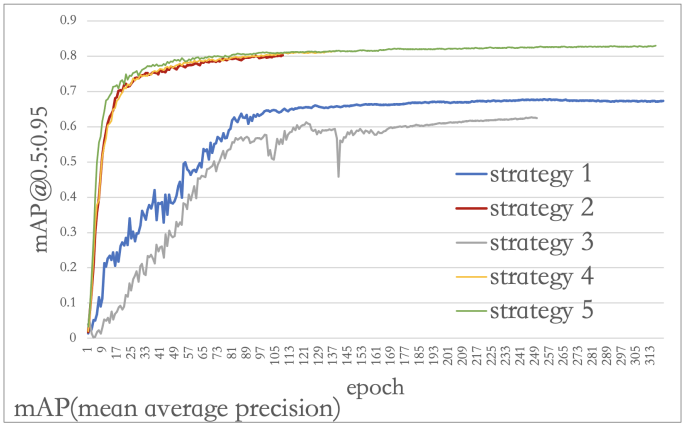


Fig. 9: LifeCLEF-2015 training mAP@0.5:0.95 of different strategies

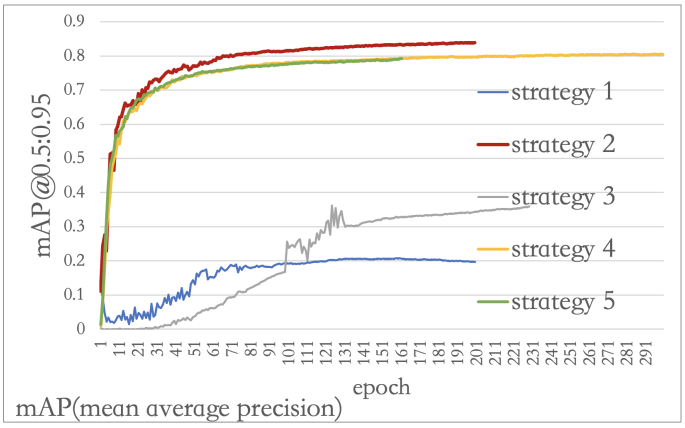


Fig. 10: Brackish training mAP@0.5:0.95 of different strategies

Take mAP@0.5:0.95 as an example, the comparison of the training process of 5 different strategies is presented in Fig 9 and Fig 10. In both datasets, Strategy 2, Strategy4, and Strategy 5 are converged significantly faster than Strategy 1 and Strategy 3, also have a higher mAP score. It is worth mentioning that in the Brackish dataset, the most difficult detect class of strategies that without Mosaic (Strategy 1 and Strategy 3) is shrimp, however, the shrimp detection performance has been increase significantly after applied Mosaic, and jellyfish becomes the hardest classification class. The challenge of detect shrimp is tiny object and move very fast. The advantages of Mosaic augmentation, accelerating training speed, and improving the effectiveness of small target detection, have been fully proved in our experimental results.

In the aspect of autoMSRCR, the two datasets have different results. On LifeCLEF-2015, the performance of Strategy 2 is lower than Strategy 1. However, on the Brackish dataset, after autoMSRCR augment, the mAP increased by one percentage point. It can be considered that autoMSRCR is valid for Brackish but invalid for LifeCLEF-2015. Combined with the features of the datasets, the reason may be that LifeCLEF-2015 was collected in shallow coral reef waters with relatively sufficient light, while the Brackish was collected in the strait, which has the challenge of low light low visibility. The autoMSRCR has a specific effect on improving low-light pictures.

Similarly, the data balance technique also has different effects on the two data sets. On the first dataset, it improves classification precision and mAP@0.5:0.95, but recall and mAP@0.5 are decreased. On the second dataset, all four indicators are worse than before balance. It can be attributed to the ineffectiveness of our data balancing Strategy. Although the balancing method we adopted increased the number of minority classes, it did not delete instances of the majority class. Even after balancing, the data set is still unbalanced.

The detailed test results of all classes of the best Strategy for the two datasets are indicated in Table VII and Table VIII, respectively. Confusion matrix and the detailed results of other strategies are in the appendix. For LifeCLEF-2015, our model detects chaetodon speculum best and detects myripristis kuntee the worst. There are not many instances of these two species, and they belong to a minority group. We speculate that the chaetodon speculum is easy to detect because of its unique shape, closer to a circle than other fishes. The whole body is yellow without other patterns; however, the myripristis kuntee is too tiny to difficult to be distinguished.

Starfish are best to be recognized for the Brackish dataset, and jellyfish are the hardest to recognize. It can be attributed to the fact that the starfish has five conspicuous feet, and it is usually stationary in the same place, easily captured by the camera, and belongs to the majority category. On the contrary, the jellyfish body is transparent, almost invisible in a low-light turbid water environment, and the speed of jellyfish is usually very fast, and there are few instances of being captured, which belong to a minority of categories.

To conclude, the proposed model has excellent performance on two underwater data sets. For different challenges that two different data poses to our model, the proposed 5 data enhancement strategies have achieved different effects. The Mosaic method effectively improves detection accuracy and accelerates training speed, and the autoMSRVCR method can improve low-light data. However, although our data enhancement method has a slight improvement in the detection effect

TABLE VII: LifeCLEF-2015 all classes best result

CLEF-2015 Mosaic + Retinex + Data Balance						
Class	Images	Labels	P	R	mAP 0.5	mAP 0.5:0.95
all	1000	1591	0.985	0.964	0.983	0.835
chaetodon trifascialis	1000	132	0.985	0.992	0.992	0.887
dascyllus reticulatus	1000	428	0.959	0.979	0.991	0.836
acanthurus nigrofuscus	1000	25	0.993	1	0.996	0.899
chaetodon lunulatus	1000	222	1	0.996	0.997	0.901
chaetodon speculum	1000	15	0.984	1	0.996	0.976
pempheris vanicolensis	1000	65	0.967	0.908	0.986	0.766
abudefduf vaigiensis	1000	20	0.994	0.9	0.98	0.799
chromis chrysura	1000	29	0.963	0.91	0.985	0.774
plectrogly-phidodon dickii	1000	127	0.96	0.969	0.981	0.762
hemigymnus melapterus	1000	20	0.995	1	0.996	0.894
myripristis kuntee	1000	15	0.994	0.867	0.866	0.601
zebrasoma scopas	1000	20	0.997	1	0.996	0.874
amphiprion clarkii	1000	88	1	1	0.997	0.841
neoglyphidodon nigroris	1000	164	0.992	0.951	0.995	0.85
dascyllus aruanus	1000	221	0.986	0.987	0.996	0.865

TABLE VIII: Brackish all classes best result

Brackish Mosaic						
Class	Images	Labels	P	R	mAP 0.5	mAP 0.5:0.95
all	1236	3432	0.985	0.975	0.987	0.83
big fish	1236	321	0.994	0.991	0.996	0.876
small fish	1236	934	0.973	0.94	0.971	0.714
crab	1236	1279	0.998	0.996	0.997	0.9148
shrimp	1236	57	0.965	0.982	0.98	0.775
jellyfish	1236	62	0.983	0.942	0.983	0.708
starfish	1236	779	0.998	0.999	0.996	0.993

of the model on LifeCLEF-2015, in Brackish, The above is counterproductive, so our data enhancement method is flawed and needs to be further improved.

VI. Conclusion

We propose an underwater fish species detection model based on deep learning, which combines data enhancement methods to perform transfer training on the pre-trained YOLOv5 model. We tested on two data sets collected in different sea areas and explored the ability of different data enhancement technologies to improve the underwater data. The contribution of this project is to verify the powerful effect of the Mosaic data enhancement method in the underwater fish dataset classify task. At the same time, we explored the enhancement effect of the autoMSRCR technology on the low-light turbid water environment and tried the data balance method for the multi-classification model. The disadvantage is that the proposed data balancing method is misdesigned, and the detection speed of YOLOv5 is not analyzed due to resource-limited. Future research will focus on the analysis detection speed of the model and the improvement of data balancing methods.

VII. Declarations

1) Declaration of Originality: I am aware of and understand the University of Exeter's policy on plagiarism and I certify that this assignment is my own work, except where indicated by referencing, and that I have followed the good academic practices.

2) Declaration of Ethical Concerns: This work does not raise any ethical issues. No human or animal subjects are involved neither has personal data of human subjects been processed. Also no security or safety critical activities have been carried out.

References

[1] J. A. Foley, K. E. Taylor, and S. J. Ghan, "Planktonic dimethylsulfide and cloud albedo: an estimate of the feedback response," *Climatic Change*, vol. 18, no. 1, pp. 1–15, 1991.

[2] E. M. Ditria, S. Lopez-Marcano, M. Sievers, E. L. Jinks, C. J. Brown, and R. M. Connolly, "Automating the analysis of fish abundance using object detection: optimizing animal ecology with deep learning," *Frontiers in Marine Science*, vol. 7, p. 429, 2020.

[3] M. Sung, S.-C. Yu, and Y. Girdhar, "Vision based real-time fish detection using convolutional neural network," in *OCEANS 2017-Aberdeen*. IEEE, 2017, pp. 1–6.

[4] D. Rathi, S. Jain, and S. Indu, "Underwater fish species classification using convolutional neural network and deep learning," in *2017 Ninth International Conference on Advances in Pattern Recognition (ICAPR)*, 2017, pp. 1–6.

[5] B. G. Weinstein, "A computer vision for animal ecology," *Journal of Animal Ecology*, vol. 87, no. 3, pp. 533–545, 2018.

[6] S. A. Siddiqui, A. Salman, M. I. Malik, F. Shafait, A. Mian, M. R. Shortis, and E. S. Harvey, "Automatic fish species classification in underwater videos: exploiting pre-trained deep neural network models to compensate for limited labelled data," *ICES Journal of Marine Science*, vol. 75, no. 1, pp. 374–389, 2018.

[7] E. S. Harvey, M. Cappo, J. J. Butler, N. Hall, and G. A. Kendrick, "Bait attraction affects the performance of remote underwater video stations in assessment of demersal fish community structure," *Marine Ecology Progress Series*, vol. 350, pp. 245–254, 2007.

[8] M. Lowry, H. Folpp, M. Gregson, and I. Suthers, "Comparison of baited remote underwater video (bruv) and underwater visual census (uvc) for assessment of artificial reefs in estuaries," *Journal of Experimental Marine Biology and Ecology*, vol. 416, pp. 243–253, 2012.

[9] A. B. Tamou, A. Benzinou, K. Nasreddine, and L. Ballihi, "Transfer learning with deep convolutional neural network for underwater live fish recognition," in *2018 IEEE International Conference on Image Processing, Applications and Systems (IPAS)*. IEEE, 2018, pp. 204–209.

[10] A. Jalal, A. Salman, A. Mian, M. Shortis, and F. Shafait, "Fish detection and species classification in underwater environments using deep learning with temporal information," *Ecological Informatics*, vol. 57, p. 101088, 2020.

[11] B. Zoph, E. D. Cubuk, G. Ghiasi, T.-Y. Lin, J. Shlens, and Q. V. Le, "Learning data augmentation strategies for object detection," in *European Conference on Computer Vision*. Springer, 2020, pp. 566–583.

[12] K. Wang, B. Fang, J. Qian, S. Yang, X. Zhou, and J. Zhou, "Perspective transformation data augmentation for object detection," *IEEE Access*, vol. 8, pp. 4935–4943, 2019.

[13] G. Jocher, A. Stoken, J. Borovec, NanoCode012, A. Chaurasia, TaoXie, L. Changyu, A. V, Laughing, tkianai, yxNONG, A. Hogan, lorenzomammana, AlexWang1900, J. Hajek, L. Diaconu, Marc, Y. Kwon, oleg, wanghaoyang0106, Y. Defretin, A. Lohia, ml5ah, B. Milanko, B. Fineran, D. Khromov, D. Yiwei, Doug, Durgesh, and F. Ingham, "ultralytics/yolov5: v5.0 - YOLOv5-P6 1280 models, AWS, Supervise.ly and YouTube integrations," Apr. 2021. [Online]. Available: https://doi.org/10.5281/zenodo.4679653

[14] C. Spampinato, R. Fisher, and B. Boom, "Image retrieval in clet-fish task," 2014.

[15] M. Pedersen, J. Bruslund Haurum, R. Gade, and T. B. Moeslund, "Detection of marine animals in a new underwater dataset with varying visibility," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 18–26.

[16] S. K. Whitmarsh, P. G. Fairweather, and C. Huveneers, "What is big bruvver up to? methods and uses of baited underwater video," *Reviews in Fish Biology and Fisheries*, vol. 27, no. 1, pp. 53–73, 2017.

[17] G. J. Edgar, N. S. Barrett, and A. J. Morton, "Biases associated with the use of underwater visual census techniques to quantify the density and size-structure of fish populations," *Journal of Experimental Marine Biology and Ecology*, vol. 308, no. 2, pp. 269–290, 2004.

[18] M. Kulbicki, "How the acquired behaviour of commercial reef fishes may influence the results obtained from visual censuses," *Journal of Experimental Marine Biology and Ecology*, vol. 222, no. 1-2, pp. 11–30, 1998.

[19] E. J. Brooks, K. A. Sloman, D. W. Sims, and A. J. Danylchuk, "Validating the use of baited remote underwater video surveys for assessing the diversity, distribution and abundance of sharks in the bahamas," *Endangered Species Research*, vol. 13, no. 3, pp. 231–243, 2011.

[20] T. B. Letessier, J. J. Meeuwig, M. Gollock, L. Groves, P. J. Bouchet, L. Chapuis, G. M. Vianna, K. Kemp, and H. J. Koldewey, "Assessing pelagic fish populations: The application of demersal video techniques to the mid-water environment," *Methods in Oceanography*, vol. 8, pp. 41–55, 2013.

[21] B. Stobart, J. A. García-Charton, C. Espejo, E. Rochel, R. Goñi, O. Reñones, A. Herrero, R. Crec'hriou, S. Polti, C. Marcos *et al.*, "A baited underwater video technique to assess shallow-water mediterranean fish assemblages: Methodological evaluation," *Journal of Experimental Marine Biology and Ecology*, vol. 345, no. 2, pp. 158–174, 2007.

[22] D. L. Watson, E. S. Harvey, M. J. Anderson, and G. A. Kendrick, "A comparison of temperate reef fish assemblages recorded by three underwater stereo-video techniques," *Marine Biology*, vol. 148, no. 2, pp. 415–425, 2005.

[23] E. Harvey, J. Butler, D. McLean, and J. Shand, "Contrasting habitat use of diurnal and nocturnal fish assemblages in temperate western australia," *Journal of Experimental Marine Biology and Ecology*, vol. 426, pp. 78–86, 2012.

[24] T. J. Willis, R. B. Millar, and R. C. Babcock, "Detection of spatial variability in relative density of fishes: comparison of visual census, angling, and baited underwater video," *Marine Ecology Progress Series*, vol. 198, pp. 249–260, 2000.

[25] S. H. Jury, H. Howell, D. F. O'Grady, and W. H. Watson III, "Lobster trap video: in situ video surveillance of the behaviour of homarus americanus in and around traps," *Marine and Freshwater Research*, vol. 52, no. 8, pp. 1125–1132, 2001.

[26] R. W. Jabado, S. M. Al Hameli, E. M. Grandcourt, and S. S. Al Dhaheri, "Low abundance of sharks and rays in baited remote underwater video surveys in the arabian gulf," *Scientific reports*, vol. 8, no. 1, pp. 1–11, 2018.

[27] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, 2019.

[28] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[29] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*. PMLR, 2015, pp. 448–456.

[30] D. Erhan, A. Courville, Y. Bengio, and P. Vincent, "Why does unsupervised pre-training help deep learning?" in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 201–208.

[31] A. S. Parihar and K. Singh, "A study on retinex based method for image enhancement," in *2018 2nd International Conference on Inventive Systems and Control (ICISC)*. IEEE, 2018, pp. 619–624.

[32] X. Fu, P. Zhuang, Y. Huang, Y. Liao, X.-P. Zhang, and X. Ding, "A retinex-based enhancing approach for single underwater image," in *2014 IEEE international conference on image processing (ICIP)*. IEEE, 2014, pp. 4572–4576.

[33] S. Zhang, T. Wang, J. Dong, and H. Yu, "Underwater image enhancement via extended multi-scale retinex," *Neurocomputing*, vol. 245, pp. 1–9, 2017.

[34] H. Yuan, S. Zhang, G. Chen, and Y. Yang, "Underwater image fish recognition technology based on transfer learning and image enhancement," *Journal of Coastal Research*, vol. 105, no. SI, pp. 124–128, 2020.

[35] T. NgoGia, Y. Li, D. Jin, J. Guo, J. Li, and Q. Tang, "Real-time sea cucumber detection based on yolov4-tiny and transfer learning using data augmentation," in *International Conference on Swarm Intelligence*. Springer, 2021, pp. 119–128.

[36] H. Huang, H. Zhou, X. Yang, L. Zhang, L. Qi, and A.-Y. Zang, "Faster r-cnn for marine organisms detection and recognition using data augmentation," *Neurocomputing*, vol. 337, pp. 372–384, 2019.

[37] N. Hassan, S. Ullah, N. Bhatti, H. Mahmood, and M. Zia, "The retinex based improved underwater image enhancement," *Multimedia Tools and Applications*, vol. 80, no. 2, pp. 1839–1857, 2021.

[38] C. Spampinato, D. Giordano, R. Di Salvo, Y.-H. J. Chen-Burger, R. B. Fisher, and G. Nadarajan, "Automatic fish classification for underwater species behavior understanding," in *Proceedings of the first ACM international workshop on Analysis and retrieval of tracked events and motion in imagery streams*, 2010, pp. 45–50.

[39] P. X. Huang, B. J. Boom, and R. B. Fisher, "Underwater live fish recognition using a balance-guaranteed optimized tree," in *Asian Conference on Computer Vision*. Springer, 2012, pp. 422–433.

[40] G. Szűcs, D. Papp, and D. Lovas, "Svm classification of moving objects tracked by kalman filter and hungarian method," in *Working Notes of CLEF 2015 Conference, Toulouse, France*, 2015.

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

[42] Z.-Q. Zhao, P. Zheng, S.-t. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 11, pp. 3212–3232, 2019.

[43] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[44] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[45] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[46] 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. 28, pp. 91–99, 2015.

[47] 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*, 2016, pp. 779–788.

[48] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[49] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[50] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[51] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[52] H. Qin, X. Li, J. Liang, Y. Peng, and C. Zhang, "Deepfish: Accurate underwater live fish recognition with a deep architecture," *Neurocomputing*, vol. 187, pp. 49–58, 2016.

[53] X. Li, M. Shang, H. Qin, and L. Chen, "Fast accurate fish detection and recognition of underwater images with fast r-cnn," in *OCEANS 2015-MTS/IEEE Washington*. IEEE, 2015, pp. 1–5.

[54] B. J. Boom, P. X. Huang, J. He, and R. B. Fisher, "Supporting ground-truth annotation of image datasets using clustering," in *Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*. IEEE, 2012, pp. 1542–1545.

[55] W. Ji, D. Liu, Y. Meng, and Q. Liao, "Exploring the solutions via retinex enhancements for fruit recognition impacts of outdoor sunlight: a case study of navel oranges," *Evolutionary Intelligence*, pp. 1–37, 2021.

APPENDIX

A. Dataset

1) LifeCLEF-2015: Fig 12 shows more sample frames of different fish species of LifeCLEF-2015 dataset, which fully reflects the challenges of the data set such as complex background, low pixels, and too small fish instances.

2) Brackish: Fig 11 shows different classes sample frames of the Brackish dataset. Obviously, it is difficult to see fish instance in this data set due to low light intensity and turbid water. At the same time, the turbidity of the water changes over time. In addition, even the same class targets also looks very different.

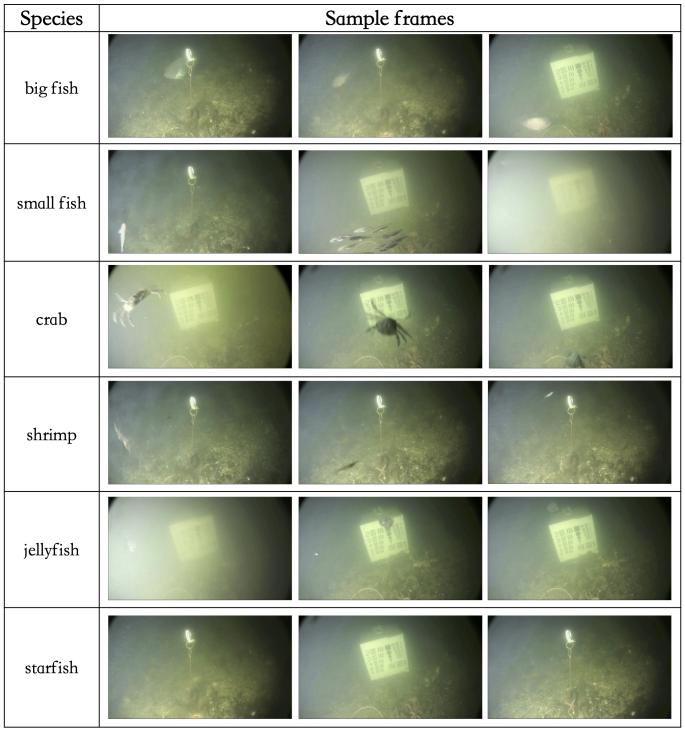


Fig. 11: Brackish sample frames of different classes

B. Detailed test results of different strategies

1) LifeCLEF-2015: The test result, confusion matrix, and PR curve of five strategies on the LifeCLEF dataset are presented in Table IX to Table XIII and Fig 13 to Fig 22, respectively.

2) Brackish: Table XIV to Table XVIII and Fig 23 to Fig 32 shown all classes test results, confusion matrix, and PR curve of different strategies on the Brackish dataset.

C. Detect sample

Fig 33 and Fig 34 are predict boxes and ground-truth samples of best strategy of LifeCLEF-2015, Fig 35 and Fig 36 are samples of the worst strategy.

Similarly, the best result and ground-truth samples of Brackish are presented in Fig 37 and Fig 38, and samples of the worst strategy predict box and the labeled box is in Fig 39 and Fig 40.

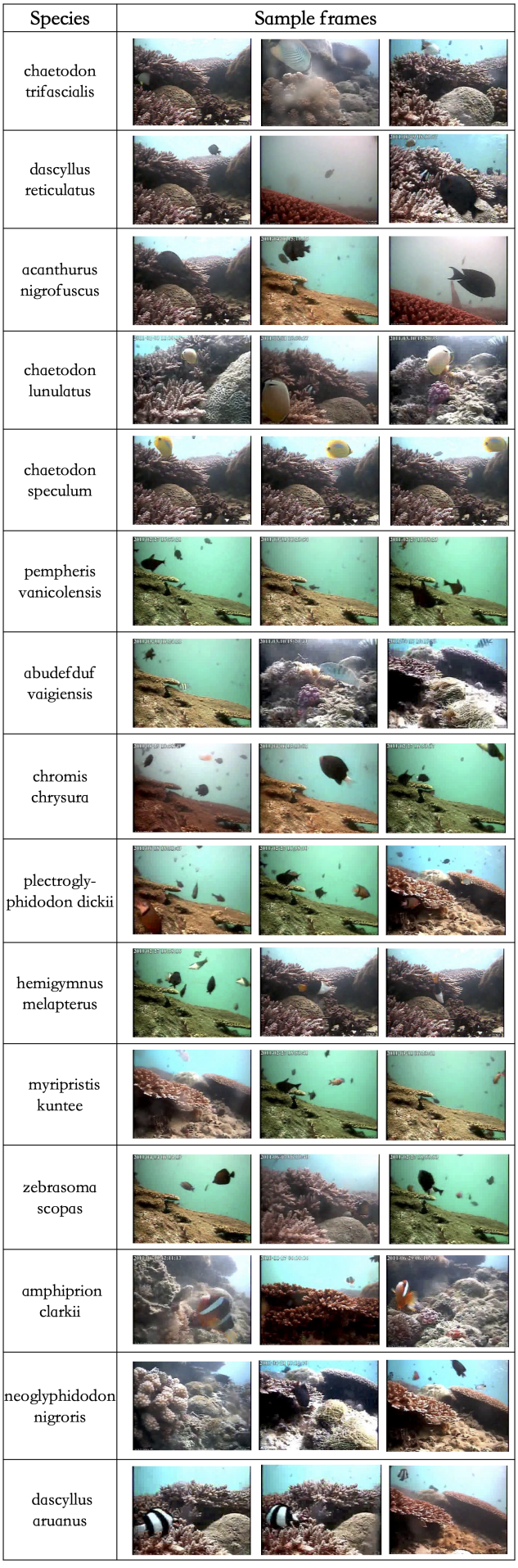


Fig. 12: LifeCLEF-2015 sample frames of different species

TABLE IX: LifeCLEF-2015 Strategy 1 test results of different class

CLEF-2015 Strategy 1: origin						
Class	Images	Labels	P	R	mAP 0.5	mAP 0.5:0.95
all	1000	1591	0.908	0.788	0.85	0.681
chaetodon trifascialis	1000	132	0.895	0.843	0.902	0.734
dascyllus reticulatus	1000	428	0.797	0.907	0.916	0.707
acanthurus nigrofuscus	1000	25	0.767	0.792	0.815	0.71
chaetodon lunulatus	1000	222	0.883	0.982	0.989	0.826
chaetodon speculum	1000	15	0.955	1	0.995	0.924
pempheris vanicolensis	1000	65	0.927	0.786	0.879	0.649
abudefduf vaigiensis	1000	20	0.895	0.6	0.633	0.496
chromis chrysura	1000	29	0.764	0.276	0.41	0.321
plectrogly-phidodon dickii	1000	127	0.956	0.717	0.783	0.598
hemigymnus melapterus	1000	20	0.92	0.8	0.9	0.742
myripristis kuntee	1000	15	1	0.714	0.836	0.593
zebrasoma scopas	1000	20	0.889	0.8	0.914	0.779
amphiprion clarkii	1000	88	0.987	0.872	0.93	0.636
neoglyphidodon nigroris	1000	164	0.992	0.781	0.88	0.719
dascyllus aruanus	1000	221	0.988	0.955	0.966	0.782

TABLE X: LifeCLEF-2015 Strategy 2 test results of different class

CLEF-2015 Strategy 2: Mosaic						
Class	Images	Labels	P	R	mAP 0.5	mAP 0.5:0.95
all	1000	1591	0.97	0.98	0.988	0.813
chaetodon trifascialis	1000	132	0.989	1	0.993	0.844
dascyllus reticulatus	1000	428	0.964	0.986	0.994	0.814
acanthurus nigrofuscus	1000	25	0.951	1	0.994	0.865
chaetodon lunulatus	1000	222	0.999	1	0.996	0.843
chaetodon speculum	1000	15	0.976	1	0.995	0.933
pempheris vanicolensis	1000	65	0.953	0.942	0.99	0.773
abudefduf vaigiensis	1000	20	0.952	1	0.995	0.755
chromis chrysura	1000	29	0.963	0.888	0.954	0.759
plectrogly-phidodon dickii	1000	127	0.932	0.961	0.978	0.727
hemigymnus melapterus	1000	20	0.982	1	0.995	0.864
myripristis kuntee	1000	15	1	1	0.995	0.712
zebrasoma scopas	1000	20	0.927	0.95	0.958	0.828
amphiprion clarkii	1000	88	0.999	1	0.995	0.828
neoglyphidodon nigroris	1000	164	0.981	0.967	0.993	0.826
dascyllus aruanus	1000	221	0.988	1	0.995	0.829

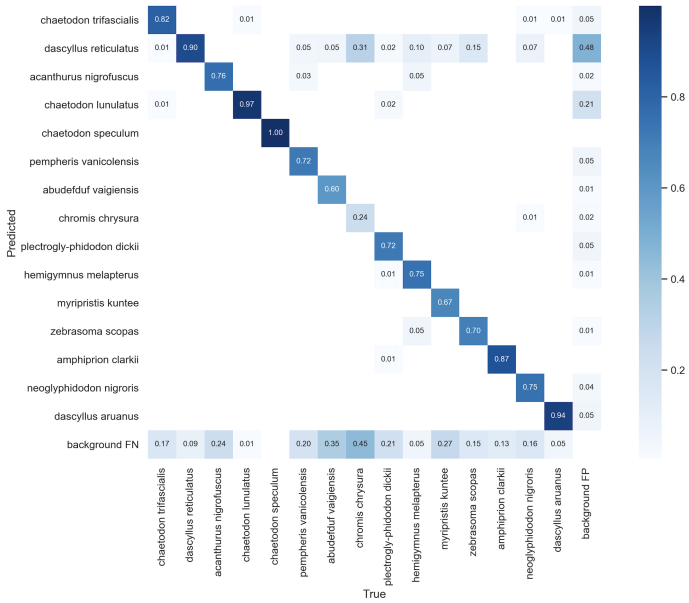


Fig. 13: LifeCLEF-2015 Strategy 1 test confusion matrix

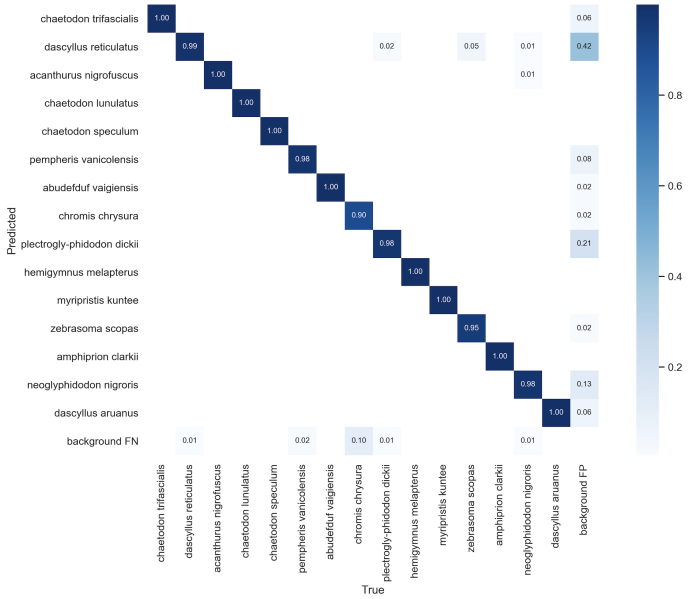


Fig. 15: LifeCLEF-2015 Strategy 2 test confusion matrix

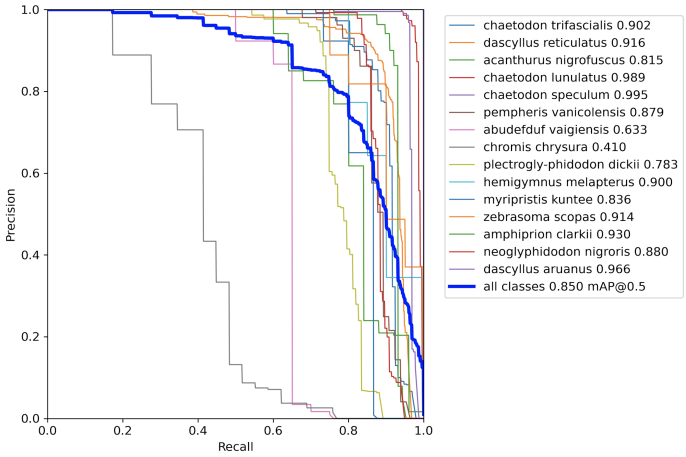


Fig. 14: LifeCLEF-2015 Strategy 1 test PR curve

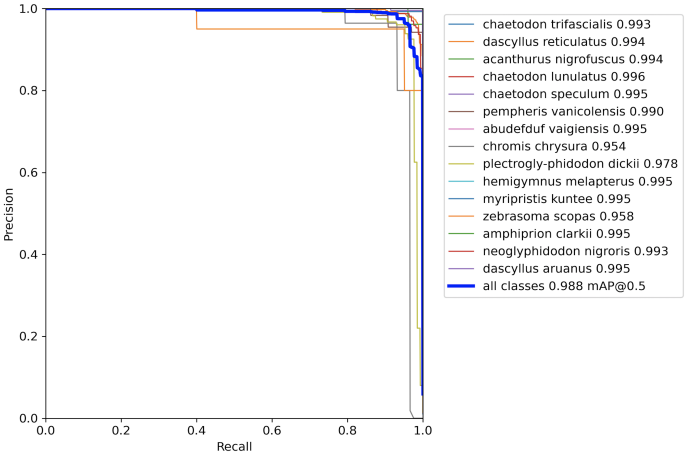


Fig. 16: LifeCLEF-2015 Strategy 2 test PR curve

TABLE XI: LifeCLEF-2015 Strategy 3 test results of different class

CLEF-2015 Strategy 3: autoMSRCR						
Class	Images	Labels	P	R	mAP 0.5	mAP 0.5:0.95
all	1000	1591	0.935	0.795	0.854	0.678
chaetodon trifascialis	1000	132	0.936	0.932	0.959	0.797
dascyllus reticulatus	1000	428	0.841	0.909	0.924	0.726
acanthurus nigrofuscus	1000	25	0.84	0.8	0.834	0.728
chaetodon lunulatus	1000	222	0.983	0.973	0.984	0.852
chaetodon speculum	1000	15	1	0.991	0.995	0.874
pempheris vanicolensis	1000	65	0.981	0.809	0.885	0.603
abudefduf vaigiensis	1000	20	1	0.474	0.638	0.483
chromis chrysura	1000	29	0.915	0.483	0.599	0.465
plectrogly-phidodon dickii	1000	127	0.898	0.761	0.799	0.592
hemigymnus melapterus	1000	20	1	0.65	0.814	0.651
myripristis kuntee	1000	15	0.838	0.933	0.94	0.644
zebrasoma scopas	1000	20	1	0.626	0.708	0.584
amphiprion clarkii	1000	88	0.906	0.773	0.843	0.6
neoglyphidodon nigroris	1000	164	0.904	0.884	0.932	0.785
dascyllus aruanus	1000	221	0.976	0.932	0.954	0.781

TABLE XII: LifeCLEF-2015 Strategy 4 test results of different class

CLEF-2015 Strategy 4: Mosaic + autoMSRCR						
Class	Images	Labels	P	R	mAP 0.5	mAP 0.5:0.95
all	1000	1591	0.983	0.975	0.992	0.815
chaetodon trifascialis	1000	132	0.976	1	0.992	0.843
dascyllus reticulatus	1000	428	0.97	0.981	0.993	0.809
acanthurus nigrofuscus	1000	25	0.984	1	0.995	0.88
chaetodon lunulatus	1000	222	0.999	1	0.996	0.849
chaetodon speculum	1000	15	0.982	1	0.995	0.955
pempheris vanicolensis	1000	65	0.984	0.937	0.991	0.748
abudefduf vaigiensis	1000	20	1	0.963	0.995	0.764
chromis chrysura	1000	29	0.964	0.937	0.968	0.732
plectrogly-phidodon dickii	1000	127	0.951	0.976	0.984	0.716
hemigymnus melapterus	1000	20	0.963	1	0.995	0.887
myripristis kuntee	1000	15	1	0.883	0.995	0.698
zebrasoma scopas	1000	20	0.996	1	0.995	0.863
amphiprion clarkii	1000	88	1	0.989	0.995	0.819
neoglyphidodon nigroris	1000	164	0.981	0.955	0.993	0.826
dascyllus aruanus	1000	221	0.991	0.999	0.995	0.832

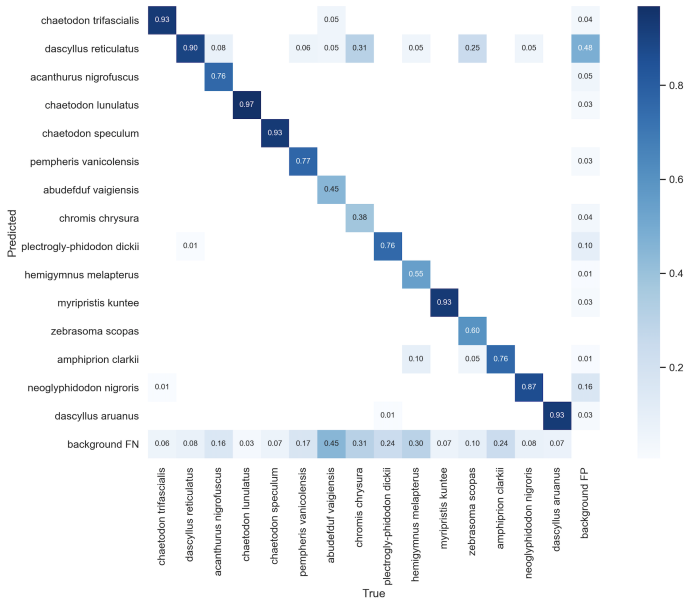


Fig. 17: LifeCLEF-2015 Strategy 3 test confusion matrix

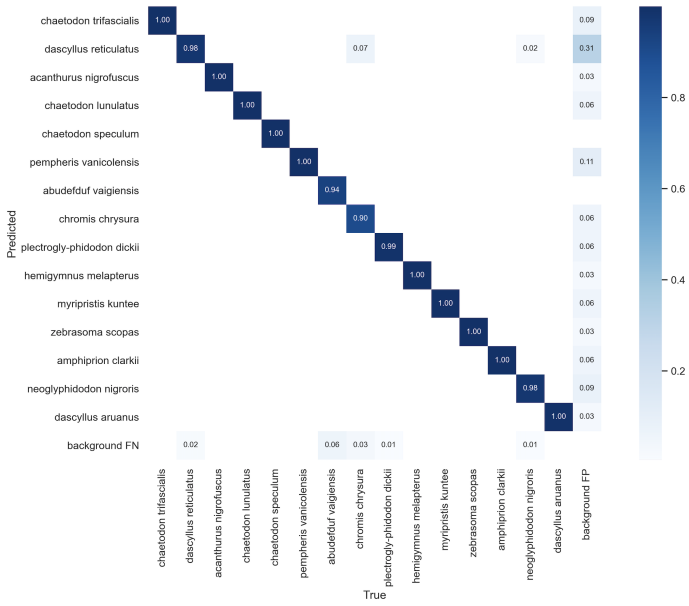


Fig. 19: LifeCLEF-2015 Strategy 4 test confusion matrix

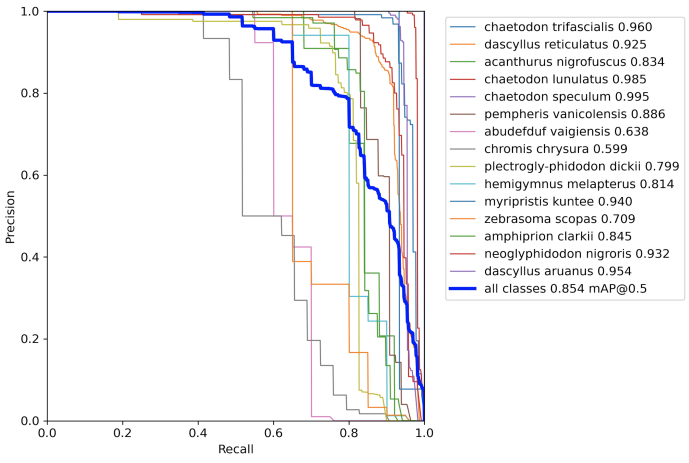


Fig. 18: LifeCLEF-2015 Strategy 3 test PR curve

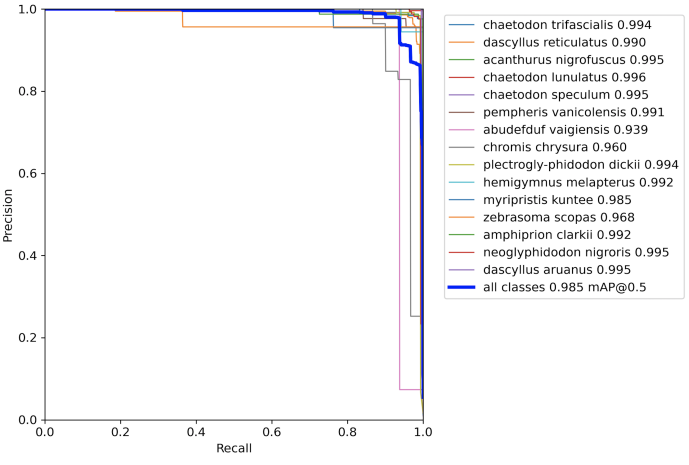


Fig. 20: LifeCLEF-2015 Strategy 4 test PR curve

TABLE XIII: LifeCLEF-2015 Strategy 5 test results of different class

CLEF-2015 Strategy 5: Mosaic + autoMSRCR + Data Balance						
Class	Images	Labels	P	R	mAP 0.5	mAP 0.5:0.95
all	1000	1591	0.985	0.964	0.983	0.835
chaetodon trifascialis	1000	132	0.985	0.992	0.992	0.887
dascyllus reticulatus	1000	428	0.959	0.979	0.991	0.836
acanthurus nigrofuscus	1000	25	0.993	1	0.996	0.899
chaetodon lunulatus	1000	222	1	0.996	0.997	0.901
chaetodon speculum	1000	15	0.984	1	0.996	0.976
pempheris vanicolensis	1000	65	0.967	0.908	0.986	0.766
abudefduf vaigiensis	1000	20	0.994	0.9	0.98	0.799
chromis chrysura	1000	29	0.963	0.91	0.985	0.774
plectrogly-phidodon dickii	1000	127	0.96	0.969	0.981	0.762
hemigymnus melapterus	1000	20	0.995	1	0.996	0.894
myripristis kuntee	1000	15	0.994	0.867	0.866	0.601
zebrasoma scopas	1000	20	0.997	1	0.996	0.874
amphiprion clarkii	1000	88	1	1	0.997	0.841
neoglyphidodon nigroris	1000	164	0.992	0.951	0.995	0.85
dascyllus aruanus	1000	221	0.986	0.987	0.996	0.865

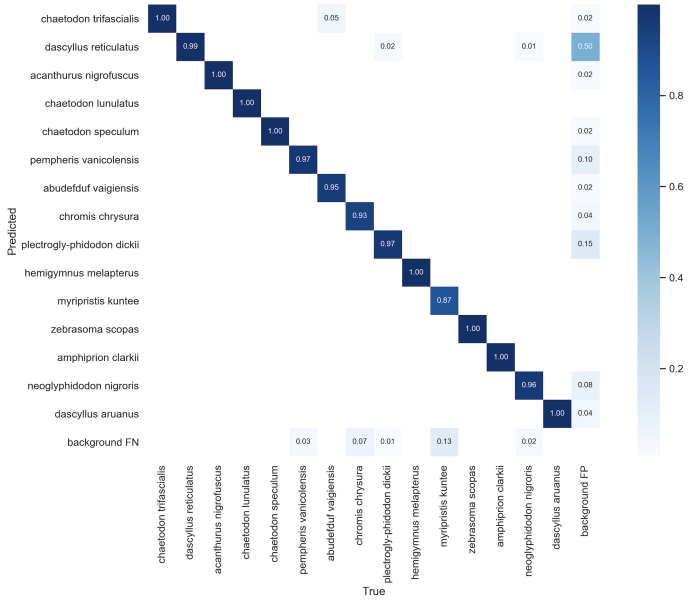


Fig. 21: LifeCLEF-2015 Strategy 5 test confusion matrix

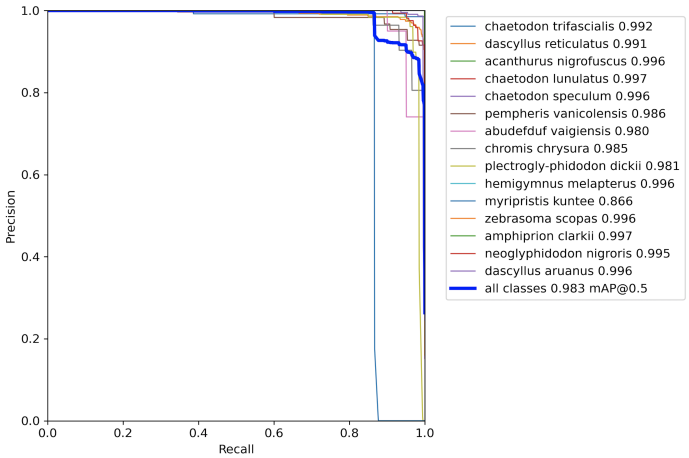


Fig. 22: LifeCLEF-2015 Strategy 5 test PR curve

TABLE XIV: Brackish Strategy 1 test results of different class

Brackish Strategy 1: origin						
Class	Images	Labels	P	R	mAP 0.5	mAP 0.5:0.95
all	1236	3432	0.656	0.375	0.428	0.205
big fish	1236	321	0.542	0.533	0.531	0.248
small fish	1236	934	0.392	0.543	0.46	0.16
crab	1236	1279	0.528	0.256	0.29	0.117
shrimp	1236	57	0.828	0.0848	0.164	0.0616
jellyfish	1236	62	0.842	0.172	0.401	0.204
starfish	1236	779	0.804	0.662	0.725	0.441

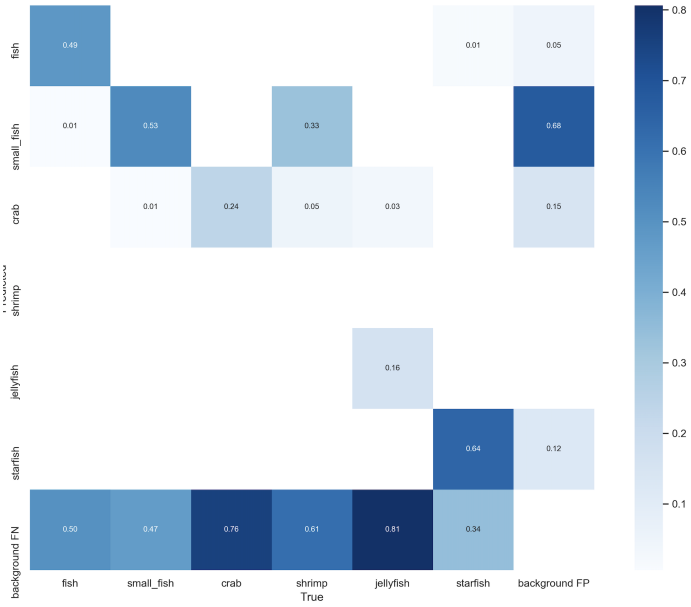


Fig. 23: Brackish Strategy 1 test confusion matrix

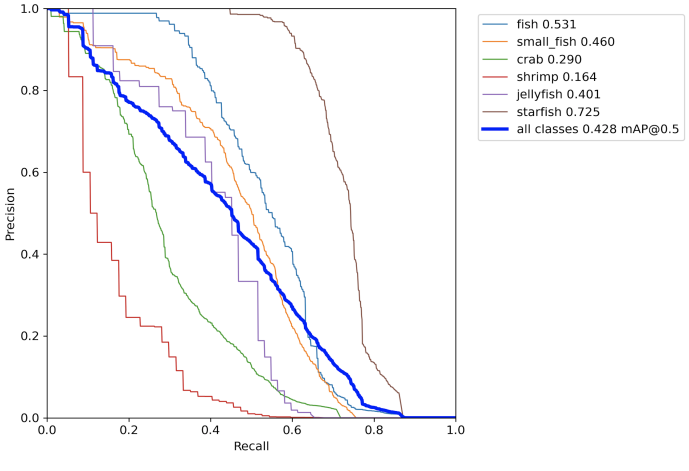


Fig. 24: Brackish Strategy 1 test PR curve

TABLE XV: Brackish Strategy 2 test results of different class

Brackish Strategy 2: Mosaic						
Class	Images	Labels	P	R	mAP 0.5	mAP 0.5:0.95
all	1236	3432	0.985	0.975	0.987	0.83
big fish	1236	321	0.994	0.991	0.996	0.876
small fish	1236	934	0.973	0.94	0.971	0.714
crab	1236	1279	0.998	0.996	0.997	0.9148
shrimp	1236	57	0.965	0.982	0.98	0.775
jellyfish	1236	62	0.983	0.942	0.983	0.708
starfish	1236	779	0.998	0.999	0.996	0.993

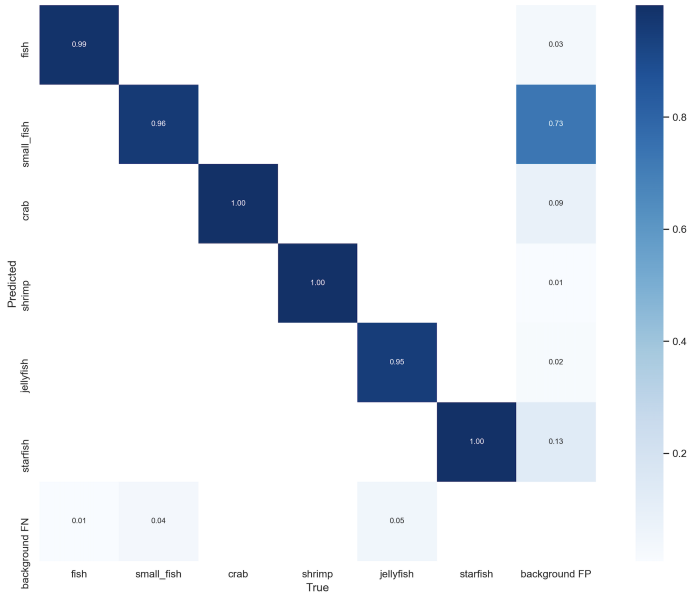


Fig. 25: Brackish Strategy 2 test confusion matrix

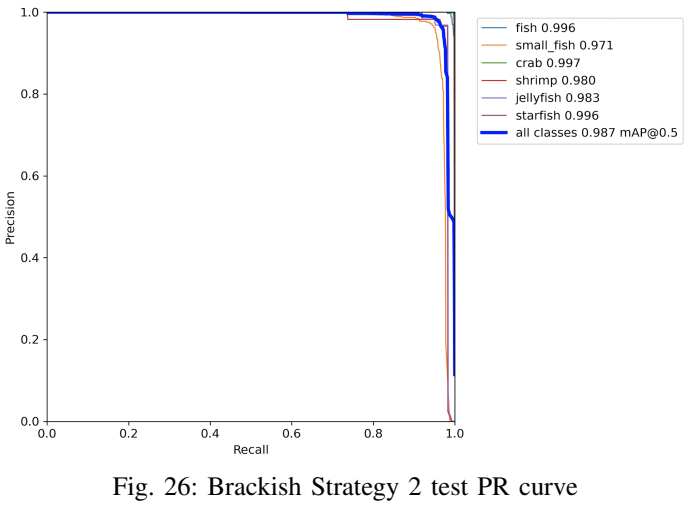


Fig. 26: Brackish Strategy 2 test PR curve

TABLE XVI: Brackish Strategy 3 test results of different class

Brackish Strategy 3: autoMSRCR						
Class	Images	Labels	P	R	mAP 0.5	mAP 0.5:0.95
all	1236	3432	0.598	0.574	0.575	0.367
big fish	1236	321	0.135	0.729	0.57	0.31
small fish	1236	934	0.538	0.581	0.552	0.235
crab	1236	1279	0.837	0.874	0.896	0.638
shrimp	1236	57	0.285	0.0877	0.127	0.0734
jellyfish	1236	62	0.799	0.257	0.376	0.207
starfish	1236	779	0.992	0.913	0.927	0.738

Fig. 27: Brackish Strategy 3 test confusion matrix

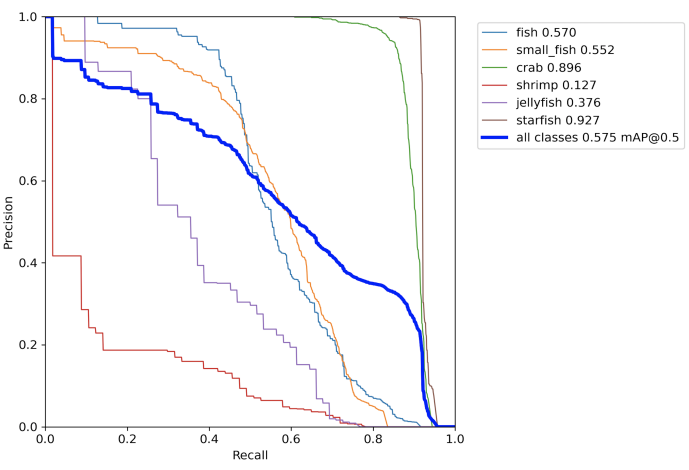


Fig. 28: Brackish Strategy 3 test PR curve

TABLE XVII: Brackish Strategy 4 test results of different class

Brackish Strategy 4: Mosaic + autoMSRCR						
Class	Images	Labels	P	R	mAP 0.5	mAP 0.5:0.95
all	1236	3432	0.981	0.966	0.984	0.756
big fish	1236	321	0.997	0.99	0.996	0.843
small fish	1236	934	0.934	0.89	0.94	0.607
crab	1236	1279	0.989	0.991	0.995	0.789
shrimp	1236	57	1	1	0.996	0.691
jellyfish	1236	62	0.966	0.928	0.98	0.681
starfish	1236	779	0.999	0.999	0.997	0.923

TABLE XVIII: Brackish Strategy 5 test results of different class

Brackish Strategy 5: Mosaic + autoMSRCR + Data Balance						
Class	Images	Labels	P	R	mAP 0.5	mAP 0.5:0.95
all	1236	3432	0.972	0.952	0.975	0.738
big fish	1236	321	0.992	0.984	0.993	0.823
small fish	1236	934	0.946	0.836	0.917	0.581
crab	1236	1279	0.986	0.988	0.995	0.76
shrimp	1236	57	0.947	0.982	0.976	0.713
jellyfish	1236	62	0.958	0.919	0.974	0.644
starfish	1236	779	0.999	0.999	0.996	0.907

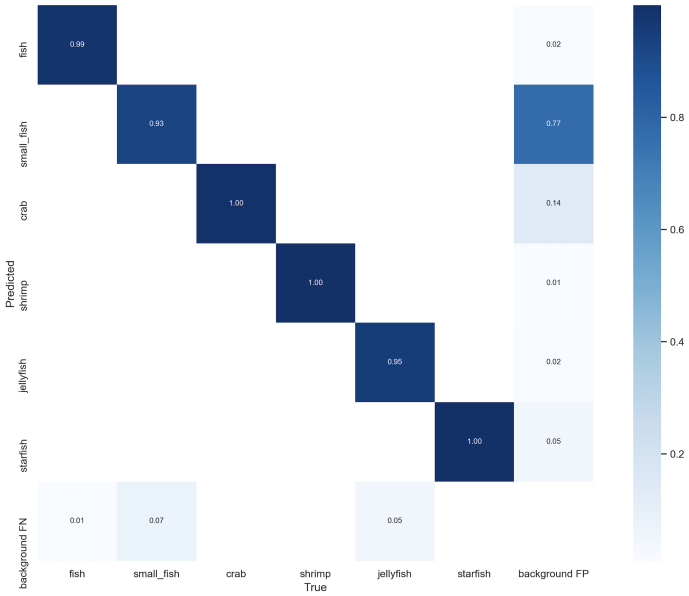


Fig. 29: Brackish Strategy 4 test confusion matrix

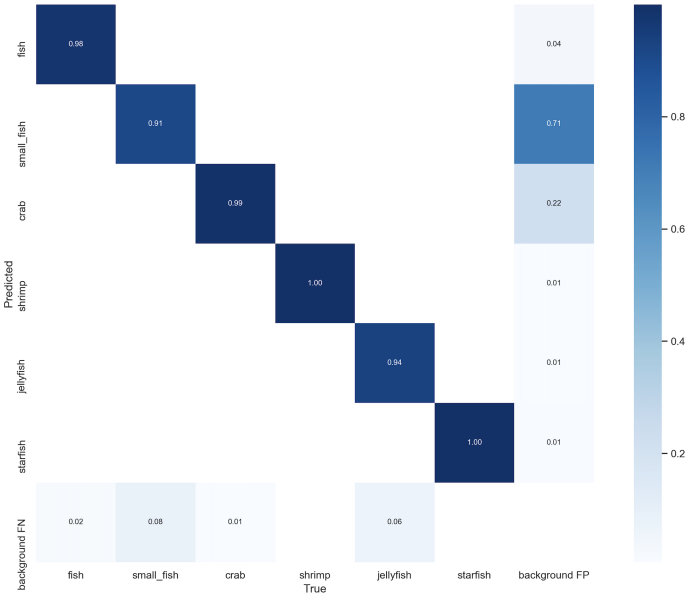


Fig. 31: Brackish Strategy 1 test confusion matrix

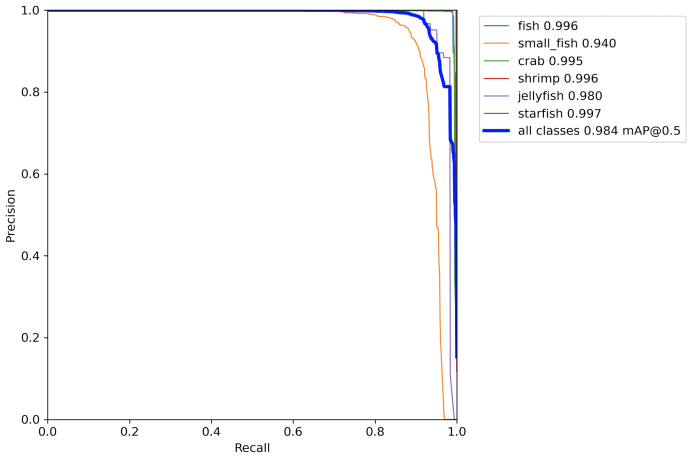


Fig. 30: Brackish Strategy 4 test PR curve

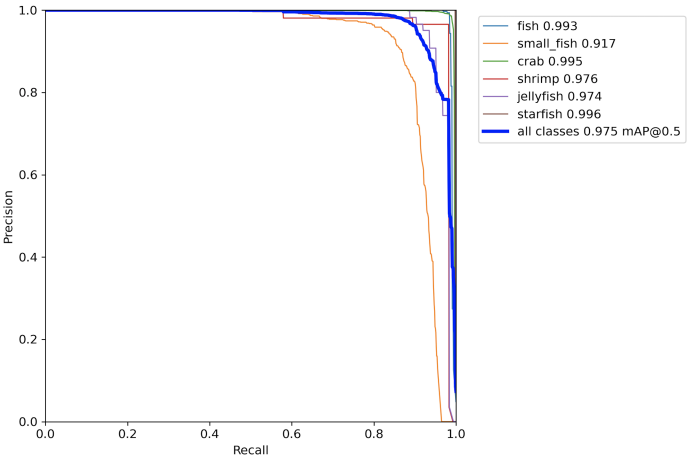


Fig. 32: Brackish strategy 5 test PR curve

Fig. 33: LifeCLEF-2015 best detect results sample (Strategy 1)

Fig. 34: Ground-truth of Fig 33

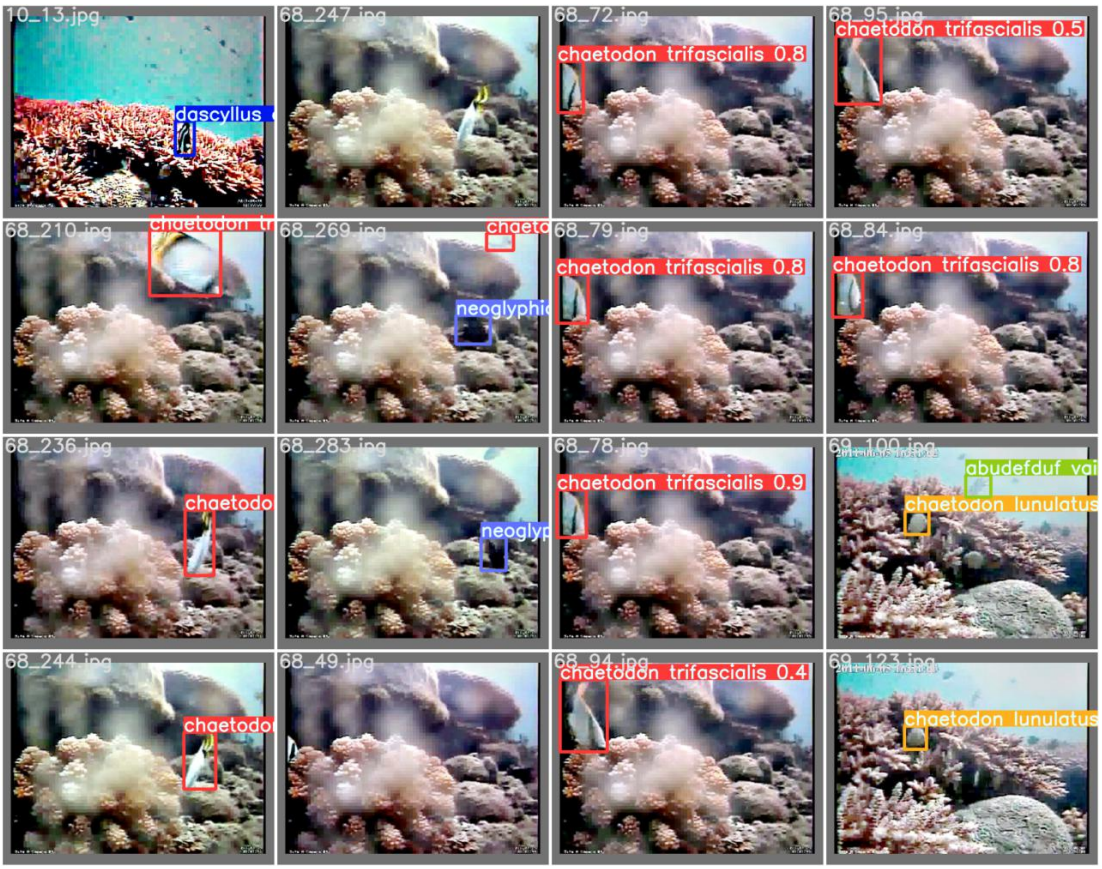


Fig. 35: LifeCLEF-2015 worst detect results sample (Strategy 3)

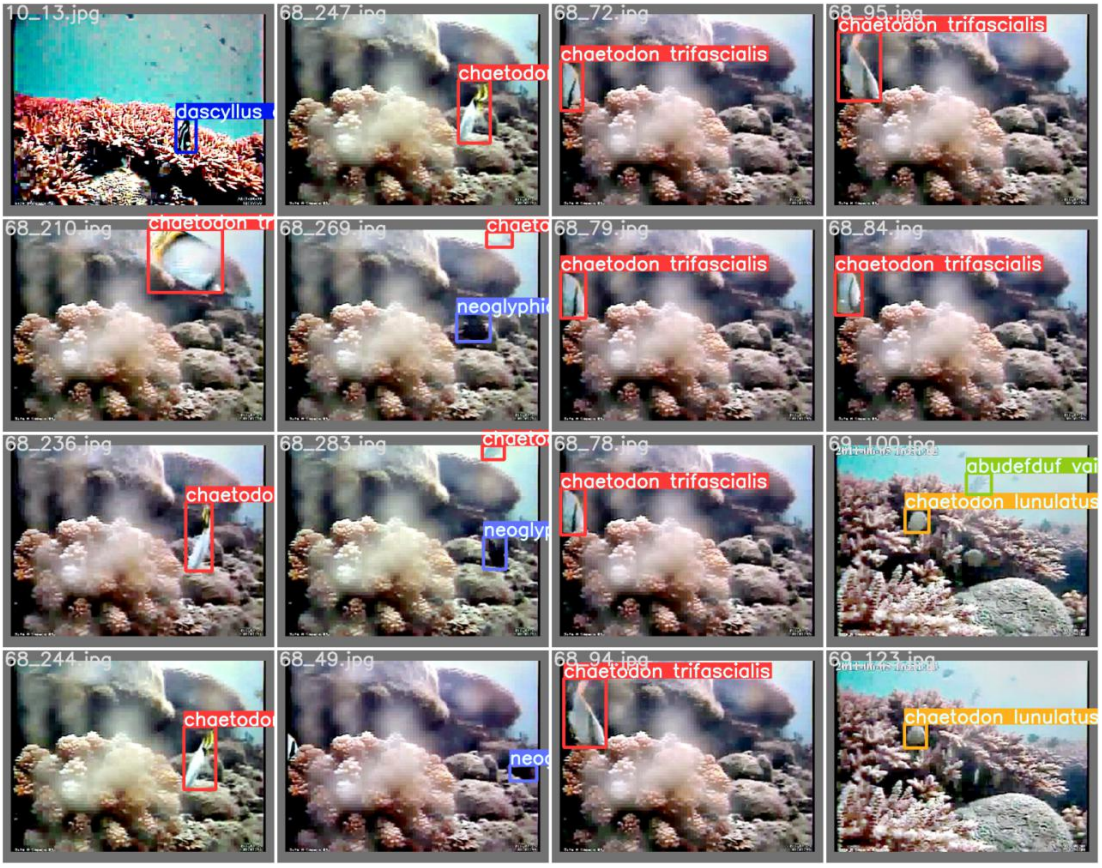


Fig. 36: Ground-truth of Fig 35

Fig. 37: Brackish best detect results sample (Strategy 2)

Fig. 38: Ground-truth of Fig 37

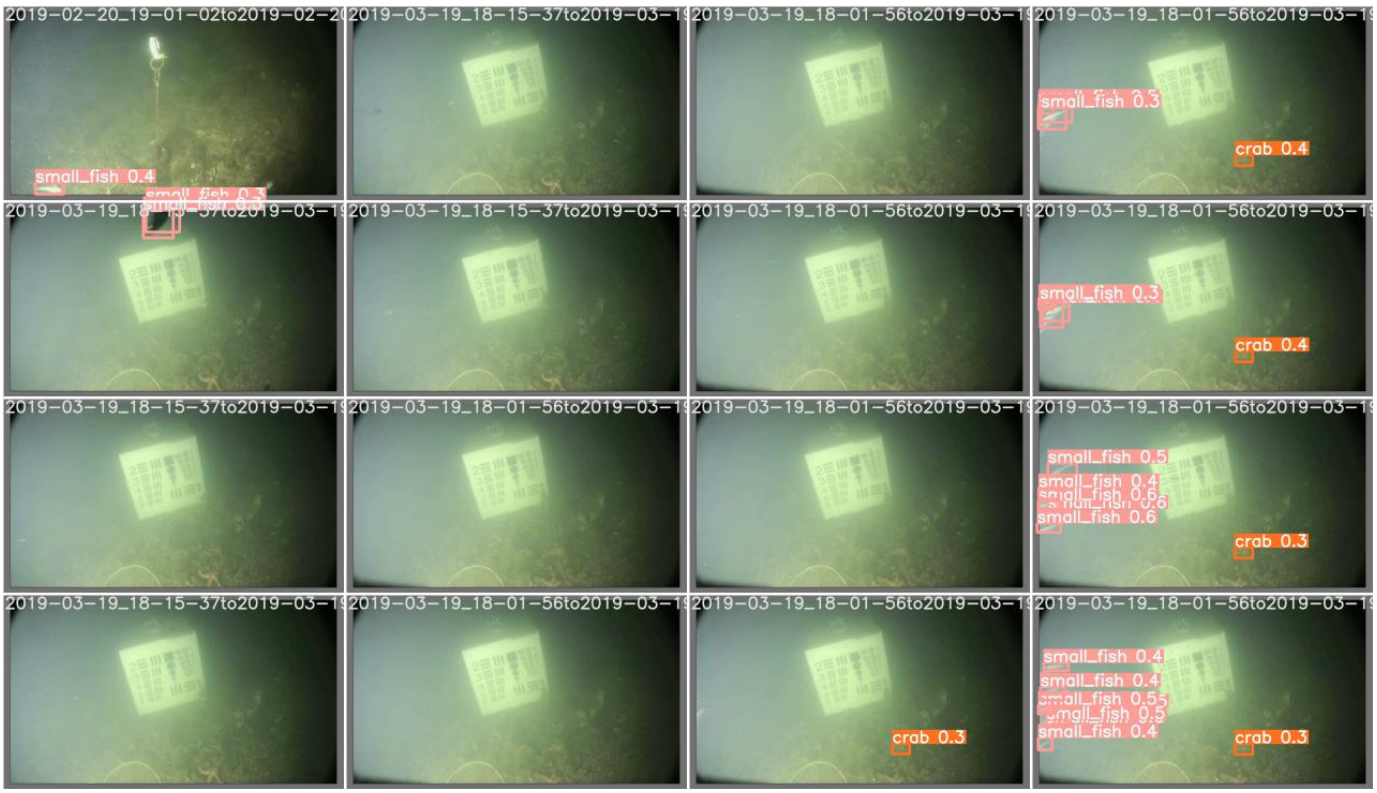


Fig. 39: Brackish worst detect results sample (Strategy 1)

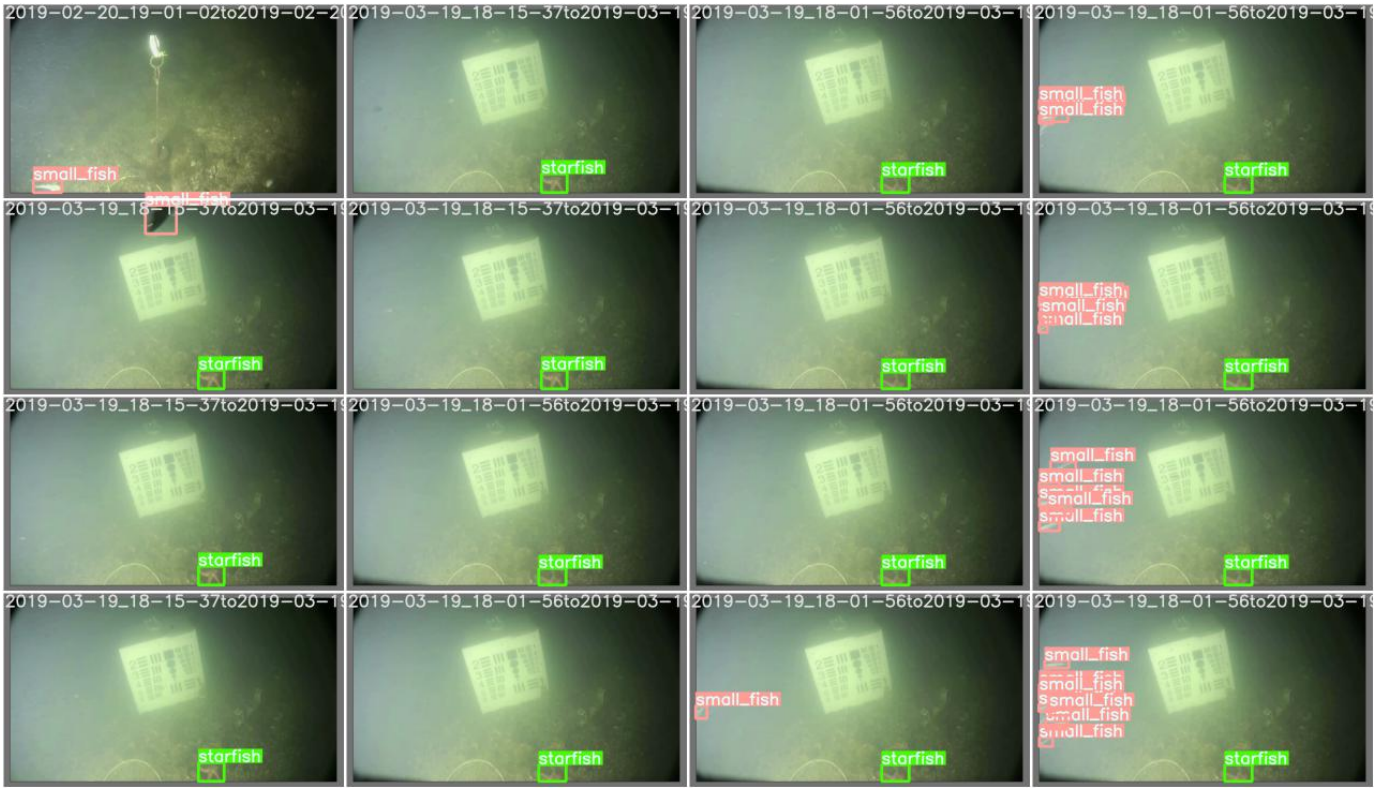


Fig. 40: Ground-truth of Fig 39