Neurocomputing 583 (2024) 127488

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Coordinate-Aware Mask R-CNN with Group Normalization: A underwater marine animal instance segmentation framework

Dewei Yi [a,*], Hasan Bayarov Ahmedov [a], Shouyong Jiang [a,d], Yiren Li [a], Sean Joseph Flinn [b], Paul G. Fernandes [c]

[a] Department of Computing Science, University of Aberdeen, Aberdeen, AB24 3UE, UK
[b] School of Biological Sciences, University of Aberdeen, Aberdeen, AB24 3UE, UK
[c] Lyell Centre, Heriot-Watt University, Edinburgh, EH14 4AP, UK
[d] School of Automation, Central South University, Changsha, 410083, China

ARTICLE INFO

Communicated by X. Gu

Keywords:
Instance segmentation
Convolutional neural network (CNN)
Underwater dataset
Generalisability

ABSTRACT

Unsustainable fishing, driven by bycatch and discards, harms marine ecosystems. Addressing this, we propose a Coordinate-Aware Mask R-CNN (CAM-RCNN) method to enhance fish detection in commercial trawls. Leveraging CoordConv and Group Normalization, our approach improves generalisation and stability. To tackle class imbalance, a compound Dice and cross-entropy loss is employed, and image data are enhanced through multi-scale retinex and colour restoration. Evaluating on two fishing datasets, CAM-RCNN excels in accuracy and generalisation, achieving the best Average Precision (AP) for instance mask and BBOX prediction in both source (39.7%, 40.2%) and target domains (24.4%, 24.2%). This method promotes sustainable fishing by selectively capturing desired fish, reducing harm to non-target species.

1. Introduction

Over the last decades, substantial improvements in fisheries management have been made to achieve better sustainability of fish stocks and global food security [1]. However, 33% of commercial marine fish stocks face overfishing [2] and even well-managed fisheries still suffer from unwanted catches, leading to discarding and bycatch [3]. When unwanted fish are caught, unnecessary time is spent sorting them from marketable fish, and those fish are then returned to the sea as discards, dead. These fish may not be counted against the permitted quota. Ocean biodiversity is also threatened if bycatch is of vulnerable species, such as marine mammals, sharks, and rays [4,5].

Fishing would be significantly more sustainable if fishing gear was more selective. Presently, regulations on the minimum size of mesh in trawl nets allow small fish to avoid capture, but there are no solutions to the problem of catching larger fish outwith quota, and bycatch. As a first step to developing more selective systems, it is vital that fish can be detected and identified prior to capture in the fishing apparatus.

Although sonar systems are used to detect fish schools and estimate the amount of fish [6], these systems are limited in their ability to identify species. They also mainly work for schooling species, which occur in midwater and do not have so much of a discard problem. In recent years, compelling progress has occurred in image processing [7–9] and object detection [10,11], which makes it possible to detect species automatically from camera images [12]. In image processing, instance segmentation can predict the pixel-wise masks and categories of instances of interest. Numerous instance segmentation methods have been proposed and state-of-the-art performance is likely to be provided by deep learning-based methods. Mask R-CNN [13] was the first deep learning-based instance segmentation method which introduces an additional branch to generate segmentation masks alongside bounding box predictions. Nowadays, instance segmentation attracts much attention from both academia and industry.

The characteristics of instance segmentation make it a promising solution to detect and identify fishes automatically. In the deep sea, the contrast of background and foreground is not salient enough and light conditions can be poor due to insufficient underwater luminosity [14]. Moreover, the images captured in deep sea often suffer from degeneration due to the noise from artificial lighting sources and different optical imaging devices [12,15], which makes it challenging for an underwater instance segmentation model to generalise well in various underwater scenes. To tackle this issue, we propose a coordinated-aware Mask RCNN (CAM-RCNN) method, to improve the accuracy and generalisation ability for detecting and identifying fish. The performance of instance segmentation generalises well in different deep-sea scenarios, under various light conditions. In particular, a coordinate-aware design with Group Normalization (GN) is incorporated into an

* Corresponding author.
E-mail address: dewei.yi@abdn.ac.uk (D. Yi).

https://doi.org/10.1016/j.neucom.2024.127488
Received 17 October 2023; Received in revised form 8 February 2024; Accepted 4 March 2024
Available online 6 March 2024

instance segmentation framework to achieve better generalisation ability. The coordinate-awareness is achieved by conducting CoordConv operations so as to provide convolutional filters with awareness of their Cartesian spatial position through adding supplementary input channels. These extra channels convey the coordinates of the data observed by the convolutional filter. For the sake of better generalisation, stability, and training efficiency, GN is integrated into our method which normalises the activations within a layer based on smaller groups of channels to reduces the dependence on large batches for effective normalisation. A compounded loss function is also proposed to improve the prediction accuracy along with image enhancement, where Dice loss is combined with Cross Entropy to handle class imbalance and improve localisation accuracy by emphasising the spatial agreement between predicted and target masks so as to contribute to better localisation of objects in segmentation tasks. Our proposed method is evaluated on two real-world underwater fishing image datasets, which contain realistic imaging in a deep-sea environment. The main contributions of our work are summarised as follows:

- To improve the generalisation ability of instance segmentation, we propose a novel Coordinate-Aware Mask R-CNN (CAM-RCNN) method by incorporating the strengths of CoordConv network and group normalisation.
- To address the issue of class imbalance and further improve the accuracy, a compounded dice and cross entropy loss is introduced to optimise our network, which can enhance detection and recognition accuracy of minority classes.
- To further boost the generalisation ability, image data are enhanced by automated multi-scale retinex with colour restoration approach during training. Inference augmentation is carried out as well during testing.
- To demonstrate the superiority of our proposed method, we conduct a comprehensive comparison against other advanced instance segmentation methods on both source domain and target domain for evaluating the accuracy and generalisation ability, respectively. In addition, an ablation study is also performed to identify the contributions of each components of our method. All experiments are conducted on our deep sea fish datasets, which were recently collected from the North Sea.

The reminder of this paper is organised as follows. Section 2 reviews the studies in instance segmentation and underwater object detection, especially for marine animals. The overall framework of our proposed CAM-RCNN method is illustrated in Section 3, where the key components of the proposed method are also introduced. Section 4 presents the experimental results of our method and other advanced methods, where an ablation study is conducted as well. Finally, we conclude this paper along with future work in Section 5.

2. Related work

2.1. Underwater object detection

The success of deep learning in computer vision [16–19], provides opportunities to apply the techniques to identify aquatic biological targets, particularly in the multi-target multi-class instance segmentation. One stream improves underwater object detection through modifying network architecture. In [14], a simple cascaded deep network is employed in fish recognition. It combines CNNs, PCA, block-wise histograms, Spatial Pyramid Pooling (SPP), with linear Support Vector Machine (SVM) and features are learned from the training data and therefore no domain knowledge of fish is required. While, it is a time-consuming model due to its large size. In order to provide pixel-wise segmentation of fish, [20] use Mask RCNN architecture to detect and segment fish simultaneously. While, the data of training and test sets are similar with each other so the generalisation ability of the Mask RCNN can be further validated. In addition, multi-domain supervision is utilised to enhance the performance of identifying fish in [21,22]. Although these approaches can increase the size of data, it may cause data imbalance. In [23], a Gaussian mixture model (GMM) and optical flow is applied to enhance the quality of the extracted feature in detecting fish so that fish biomass and the assemblage can be monitored in water bodies. One challenge of this method is that the parameters of GMM need to be carefully tuned for the sake of balancing the rate of false alarm and misdetection. Another steam improves underwater object detection by employing better data prepossessing. [12] uses the Faster R-CNN method with data augmentation to identify marine species, which shows promising result when training and testing images come from the same source. However, the diversity of augmented images are insufficient, which may lead to inadequate generalisation ability when training and testing images come from different domains. To segment aquatic images in poor visibility, various image enhancing techniques are integrated in [24], such as gamma correction and sharpening, to enhance segmentation performance in such underwater circumstances. While only a small number of images are tested. In addition, [25] presents a novel automated MSRCR approach for image enhancement which can be integrated into the instance segmentation framework. Given its promising performance, this image enhancement method is adopted into our proposed method. So far, most of existing work in underwater object detection is conducted in coastal water. Different from the coastal water, marine animal detection in deep sea has its unique challenges. For example, fishes move freely and quickly in 3-D space and they tend to hide behind other fishes [26]. In addition, we needs artificial light source to visualise fish in deep sea and energy loss during the propagation of light diminishes its intensity [27], which introduce extraneous noises and bring the challenge for generalisation ability.

To date, insufficient attention has been applied to the generalisation ability of instance segmentation in an underwater environment, where a model can perform well in the source domain but underperform in the target domain. Therefore, in this paper, we focus on the generalisation ability.

2.2. Marine animal segmentation and instance segmentation

Marine animal study has gained increasing research attention, which raise significant demands for fine-grained marine animal segmentation techniques [27]. In recent years, advance computer vision techniques have been applied to various marine animal related studies, including fish identification [26], marine animal monitoring [28], and underwater image enhancement [29], etc. Among these topics, marine animal segmentation plays a vital role, which can provide important information for identifying marine animals from underwater scenes. Such information has great potentials for the fishery industry to conduct more effective monitoring of the fishery resources. Different from underwater object detection which identifies the bounding box on a underwater object and label its category, marine animal segmentation assigns pixel-wise prediction in an image. To achieve the sustainable fishing, we need to treat multiple fish of the same specie as distinct individual instance and therefore it is not enough to solely carry out underwater object detection or marine animal segmentation. To solve this problem, instance segmentation is introduced in this work as discussed below.

Instance segmentation is the task of unifying object detection and semantic segmentation, which is used to predict the pixel-wise masks and generate the bounding box for categories of instances of interest. Mask RCNN [13], the first instance segmentation networks which introduces an additional branch to generate segmentation masks alongside bounding box predictions. Inspired by Mask RCNN [13], many advanced instance segmentation methods [30–32] have been proposed to improve the performance of object detection and segmentation. CenterMask [30] is a simple and effective anchor-free framework,

which extends an anchor-free one-stage object detector with a spatial attention-guided mask (SAG-Mask). With the spatial attention map, the SAG-Mask produces a segmentation mask for each detected box in order to identify relevant pixels and suppress noise. Such a design imrpoves inference speed while reduces inference speed. CondInst [31] leverages dynamic instance-aware structures, where network parameters are adapted based on the instance to be predicted rather than instance-wise RoIs. Without RoI operations, CondInst can generate a high-resolution instance mask with edges. This works when training on the large-scale data and may not work very well when only a small-scale data is available. SOLOv2 [32] is an improved version of SOLO, which segments objects based on locations. SOLOv2 performs dynamic instance segmentation on an input image without detecting BBOXs. Its object mask contains mask kernel prediction and mask feature learning to produce convolution kernels and feature maps, respectively.

Although these instance segmentation methods have been successfully used in autonomous driving, pedestrian detection, and crowd analysis, etc., there is insufficient work done in underwater instance segmentation, especially in deep sea fishing. In the deep sea, the contrast of background and foreground is not salient enough and light conditions can be poor.

3. Coordinate-Aware Mask R-CNN (CAM-RCNN)

This section introduces the details of our proposed CAM-RCNN for underwater instance segmentation, which covers the main characteristics of the proposed method. First, the overall framework of CAM-RCNN is provided in Section 3.1. Second, we enforce coordinate-awareness by introducing CoordConv and group normalisation is used to speed up convergence. Consequently, coordination information can be obtained to locate fishes. The detailed description of coordinate convolution and group normalisation are provided in Section 3.2. Third, the loss function of our proposed method are presented in Section 3.3. Fourth, our proposed instance Non-Maximum suppression (NMS) is described in Section 3.5. Moreover, inference augmentation is adopted in our proposed method as well to further improve the accuracy and speed.

3.1. Framework of CAM-RCNN

The overview framework of our proposed CAM-RCNN is provided in Fig. 1. Our method enhances the generalisation ability and accuracy through three stages: pre-processing, network learning, and post-processing. In pre-processing, we enhance the quality of raw images by using automated multi-scale retinex with colour restoration (AMSRCR) approach. That is, an input raw underwater fish image is enhanced by AMSRCR, where some examples of raw image and enhanced images are provided in Figs. 3 and 4. In network learning, CoordConv layer and Group Normalisation are integrated into our framework to accelerate convergence, and enhance the generalisation in unseen data, respectively. Moreover, a compound dice cross entropy loss is introduced to mask loss so as to guide network optimisation. Such a design alleviates the class imbalance problem and further improve the performance. In post-processing, Inference Augmentation is adopted to enhance accuracy, where predictions are conducted on both the original test image and its augmented images. All of the modifications introduced in CAM-RCNN are motivated by the enhance generalisation ability and prediction performance. More specifically, our proposed CAM-RCNN method utilises the CoordConv layer as the final convolutional layer in the mask head FCN. A GN layer is introduced after each convolutional layer in the mask branch FCN. Furthermore, we propose a compound dice and cross-entropy loss to further boost the performance. The alterations we apply replicate some of SOLOv2's novel architecture components onto the baseline Mask RCNN instance segmentation framework through merging. More specifically, we analyse the innovative parts of SOLOv2 to identify which would have a positive impact on generalisability. Finally, we decide to alter the mask head of the classic Mask RCNN model by: changing its deepest convolutional layer to CoordConv [33]; adding a GN data normalisation layer after each convolutional (excluding the deconvolutional) layer in the FCN; and replacing the default mask loss function with the compound DBL loss function. Furthermore, we introduce the novel Matrix Bounding Box Non-Maximum Suppression (MBBNMS) technique in the region proposal branch of CAM-RCNN which imitates the Matrix NMS algorithm of SOLOv2. Then, we delve into the implementation and advantages of the SOLOv2 components and incorporate in CAM-RCNN.

3.2. Coordinate-awareness with group normalisation (CAGN)

3.2.1. CoordConv

CoordConv extracts more precise spatial information by adding auxiliary channels containing coordinate information. This addresses the imprecise location issue caused by zero padding [34]. In particular, these auxiliary channels are concatenated to an convolutional layer, where the operations of two coordinates, i and j, are added. Concretely, the i coordinate channel is a $h \times w$ matrix with the rank of one. The matrix's first row is filled by zeros, the second row is filled by ones, and the third row is filled by twos, and so on. To the analogy, the j coordinate channel is filled by rows with constant values. Since we conduct convolution in 2D, two (i, j) coordinates are able to fully define an input pixel. With considering the extendibility, our method also allows for the insertion of an extra channel denoting the r coordinate. The coordinate channels of C_i, C_j, and C_r can be denoted as follows.

$$C_i = \begin{bmatrix} 0 & 0 & \dots & 0 \\ 1 & 1 & \dots & 1 \\ \dots & \dots & \dots & \dots \\ n & n & \dots & n \end{bmatrix} \quad C_j = \begin{bmatrix} 0 & 1 & \dots & n \\ 0 & 1 & \dots & n \\ \dots & \dots & \dots & \dots \\ 0 & 1 & \dots & n \end{bmatrix} \tag{1}$$

$$C_r = \sqrt{(C_i - h/2)^2 + (C_j - w/2)^2}$$

where h and w are the height and width of the feature map. For both i and j coordinate, a final linear scaling is also conducted to normalise the value within the range of $[-1, 1]$. According to [35], it finds that more than one CoordConv layers do not deliver notable improvement and therefore a single CoordConv layer is sufficient for spatially variant/position sensitive predictions. Hence, we modify just a single convolutional layer in the mask head of our architecture.

3.2.2. Group normalisation

It is commonly known that normalising the inputs speeds up training [36], makes optimisation easier, and allows extremely deep networks to converge. The general formulation of feature normalisation can be given by:

$$\hat{x}_i = \frac{1}{\sigma_i}(x_i - \mu_i), \quad \mu_i = \frac{1}{m}\sum_{k \in S_i} x_k, \quad \sigma_i = \sqrt{\frac{1}{m}\sum_{k \in S_i}(x_k - \mu_i)^2 + \epsilon} \tag{2}$$

where x_i is the feature computed with ith index. Given a image, $i = (i_N, i_C, i_H, i_W)$ is a 4D vector indexing the features in (N, C, H, W). N is the batch axis, C is the channel axis, and H and W are the spatial height and width axises. In addition, μ_i and σ_i are the mean and standard deviation (std) of ith index. ϵ is a constant term with a small value. S_i is the set of pixels, where the mean and std are calculated and m is the size of the set. In a GN layer, the set of S_i is defined as follows.

$$S_i = \{k | k_N = i_N, \lfloor \frac{k_C}{C/G} \rfloor = \lfloor \frac{i_C}{C/G} \rfloor\} \tag{3}$$

where G is the number of groups, which is a pre-defined hyper-parameter. C/G is the number of channels per group. $\lfloor \bullet \rfloor$ is the floor operation so "$\lfloor \frac{k_C}{C/G} \rfloor = \lfloor \frac{i_C}{C/G} \rfloor$" represents that the indexes i and k are in the same group of channels, assuming each group of channels are stored in a sequential order along the C axis. GN computes μ and σ based on

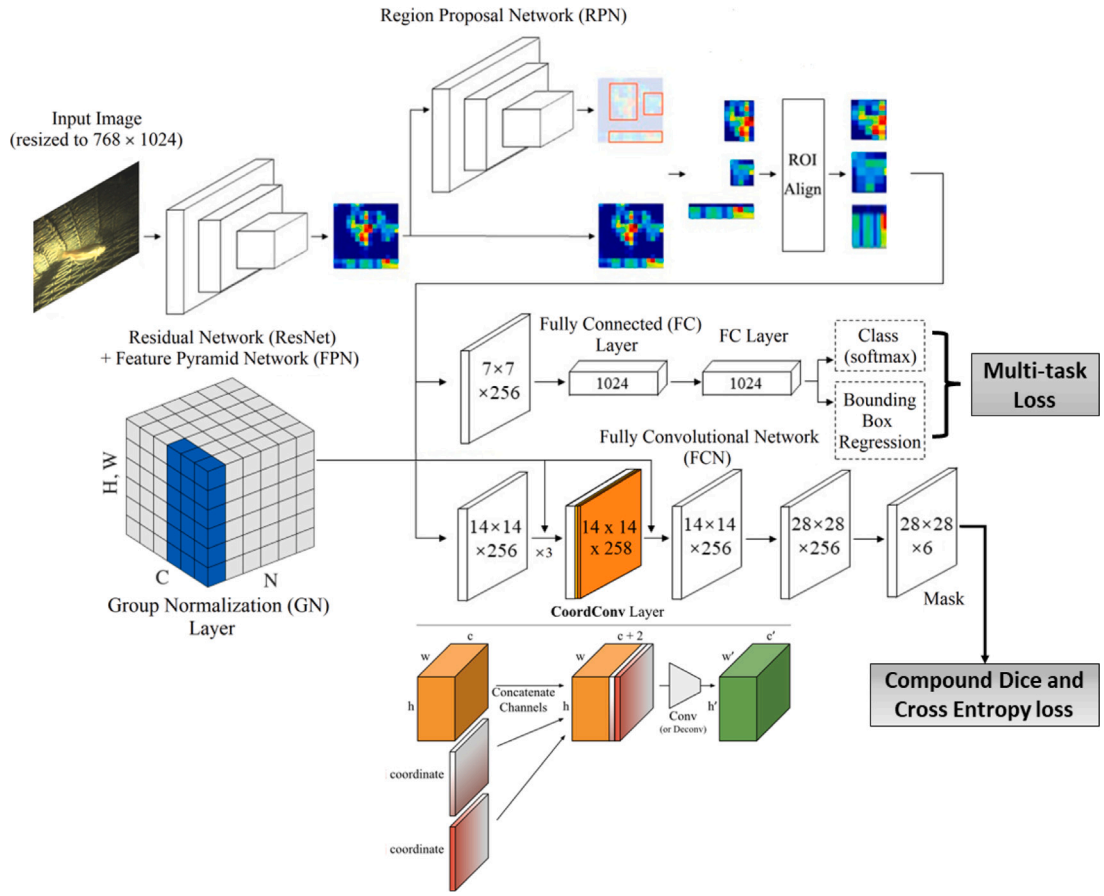

Fig. 1. The overall framework of the proposed CAM-RCNN model. Opposed to the MRCNN model, CAM-RCNN adopts: the novel MBBNMS applied on the region proposals of the RPN; a CoordConv as the last convolutional layer in the FCN of its mask prediction branch; and 4 GN layers after each convolutional layer in the mask head FCN.

the (H, W) axes and a group of $\frac{C}{G}$ channels. Given S_i in Eq. (3), the pixels in the same group are normalised together by the same μ and σ in a GN layer.

Here, we introduce the GN layer to our CAM-RCNN model in an attempt to achieve faster training and inference times, reduced error rate and a better generalisation performance. In summary, GN is a group-wise normalisation that works by grouping channels and normalising the characteristics inside every group, which improves accuracy and stability over BN-based method as mentioned in [36]. Compared to BN, GN does not use the batch dimension and therefore does not affect by batch size. In our method, the employed GN layers divide the channels into groups of 32.

3.3. Loss function

The overall loss includes two main kinds of losses: mask loss and multi-task loss. For the mask loss, a compound dice and cross entropy loss is proposed to produce fine instance masks. For multi-task loss, it can be further separated into classification loss and BBOX loss. The detailed description for different types of losses are provided as follows.

3.3.1. Mask loss

The prediction of instance mask is implemented by using a compound loss function, which combines dice loss and cross entropy loss together. The compound dice and cross entropy loss is named DCE, which achieves a synergy effect on improving generalisation ability and accuracy. In our network, DCE function is used to compute mask loss as below.

$$\begin{aligned} L_{mask} &= L_{DCE}(y, \hat{y}) \\ &= \frac{y + \hat{y} - 2y\hat{y}}{y + \hat{y} + \epsilon} - y\log(\hat{y}) + (1 - y)\log(1 - \hat{y}) \end{aligned} \tag{4}$$

where y is ground truth of mask and $\hat{y}$ is the predicted mask. In addition, ϵ is the smoothing factor, which is set to e^{-7}. For mask loss, there is a prediction mask produced for each class and therefore the L_{mask} loss is specified as the average binary cross entropy loss for segmentation. In addition, dice loss is employed due to its efficiency and consistency for instance mask prediction during training.

3.3.2. Multi-task loss

The multi-task loss can be further divided into classification loss (L_{cls}) and BBOX loss (L_{bbox}). For each training RoI, it is labelled with a ground-truth class u and a ground-truth BBOX regression target v. A multi-task loss on each labelled RoI to jointly train for classification and BBOX regression is given by

$$\begin{aligned} L(p, u, t^u, g^u) &= L_{cls} + L_{bbox} \\ &= L_{cls}(p, u) + L_{bbox}(t^u, g^u) \\ &= -\log p_u + \sum_{i \in x,y,w,h} L_1^{smooth}(t_i^u - g_i^u) \end{aligned} \tag{5}$$

where p is the predicted class of a given RoI and u is its true class. p_u is the probability of true class u. The BBOX loss L_{bbox} is defined over the ground truth of BBOX for class u, $g^u = (g_x, g_y, g_w, g_h)$ and a predicted BBOX $t^u = (t_x^u, t_y^u, t_w^u, t_h^u)$. For the BBOX regression, the smooth L_1 loss is used as shown in Eq. (6).

$$L_1^{smooth}(Err) = \begin{cases} 0.5(Err)^2 & if\ |Err| < 1 \\ |Err| - 0.5 & otherwise \end{cases} \tag{6}$$

where L_1^{smooth} is smooth L_1 loss, which is less sensitive to outliers. Err is the error between predicted BBOX t^u and the ground truth of BBox g^u.

3.4. Image enhancement

It is common that there is a discrepancy exists between recorded colour images and the direct observation of underwater scenes. To tackle the issue, Automated Multi Scale Retinex with Color Restoration (AMSRCR) approach [25] is integrated into our proposed method to enhance the image quality and model generalisation ability, which can automatically determine the higher and lower clipping points for MSRCR, as shown in Eq. (7), by using variance of the histogram and occurrence frequency of pixels.

$$R_i(x,y) = G[\beta \log[\alpha I_i'(x,y)](\log[I_i(x,y)] - \log[F(x,y) * I_i(x,y)]) + b] \quad (7)$$

where $I_i(x,y)$ is the image distribution in the ith colour spectral band. $F(x,y)$ is the surround function and $R_i(x,y)$ is the associated retinex output. β is a gain constant and α is a factor to control non-linearity strength. In addition, the value of final gain is denoted as G and the value of offset is denoted as b, where β, α, b, and G are set to 46. 125, −30, 192, respectively. The examples of AMSRCR enhanced image is presented in the second column of Figs. 3 and 4.

3.5. Instance NMS for inference augmentation

To deliver both mask and BBOX detections, we integrate BBOX into Matrix NMS in our method to produce instance NMS. More specifically, we first utilise the "coordinate trick" strategy [13] to perform NMS independently per class. Then, we add an offset to all the boxes. The offset is only dependent on the class, which is large enough to ensure boxes from different classes do not overlap with each other. Next, we sort the scores of the predicted BBOXs in descending order and also the BBOXs as per their sorted scores. After that, we use IoU for BBOXs instead of the mask IoU of Matrix NMS to derive our instance NMS.

Our instance NMS plays the role to suppress a predicted mask along with combining inference augmentation. The decay factor of the predicted mask is affected by two aspects. For the sake of brevity, we denote the predicted mask as m_j. The first aspect is that m_j is applied the penalty when the confidence score s_i of prediction m_i is greater than the confidence score s_j of prediction on m_j. The penalty of each prediction m_i on m_j could be easily computed by $f(iou_{i,j})$. The second aspect is the suppressed probability of m_i. Since the suppressed probability of m_i is not easy to be computed directly, we approximate the probability by the most overlapped prediction on m_i by Eq. (8) due to positive correlation between the suppressed probability and the IoUs.

$$f(iou_{\_,i}) = \min_{\forall s_k > s_i} f(iou_{k,i}),\ f(iou_{i,j}) = \exp(-\frac{iou_{i,j}^2}{\sigma}) \quad (8)$$

$$decay_j = \min_{\forall s_i > s_j} \frac{f(iou_{i,j})}{f(iou_{\_,i})},\quad s_j = s_j \bullet decay_j \quad (9)$$

where s_i and s_j are the confidence scores of predictions m_i and m_j, respectively. The final decay factor and the updated score are computed by Eq. (9).

For inference augmentation, let I be a given input image and τ be a transformation operation. If one chooses $\tau = \{\tau_1, \tau_2, \ldots, \tau_{|\tau|}\}$ as a candidate set of augmentations and the inference augmentation can be formulated as follows.

$$y_{IA} = \frac{1}{|\tau|} \sum_{i=1}^{|\tau|} F_{NMS}[\Theta_{target}(\tau_i(I))] \quad (10)$$

$$F_{NMS}[Ins_j] = Ins_j;\ if\ f(iou_{i,j}) < s_j \quad (11)$$

where Θ_{target} is the neural network to generate predicted masks and BBOXs. Ins_j represents the information of mask and BBOX for jth instance.

4. Experimental evaluation

4.1. Datasets

In the dataset, the aquatic images are collected from different deployments conducted in the North Sea, which are named Sparkling Star and Shetland deployments. The former consists of a single deployment carried out in 2019. The latter is a more recent collected images in 2022. The images of Sparkling Star dataset contain aquatic animal instances amounting to 638, where there are 502 instances for source domain train and 136 instances for source domain test, correspondingly. The images of Shetland dataset includes 156 instances for target domain test. The example of instance and images are presented in the first and second columns of Fig. 4. There are six categories of aquatic animals included in the Sparkling Star and Shetland deployment datasets: Cod, Dogfish, Flatfish, Decapod, Squid, and Jelly. In our datasets there are 409 images and 103 images from source domain used for source train and source test, respectively. For the target test set, the aquatic images of target test set are completely unseen and look quite different compared to source train set, which is used to demonstrate the generalisation ability of our method. Some examples of Cod, Dogfish, Flatfish, Decapod, Squid, and Jelly are presented in Fig. 2 for source domain images and target domain images.

In addition, data augmentation is used to increase the number of data for training and testing so as to help perform regularisation and prevent overfitting. For data augmentation in training time, it is frequently used with picture data, where the replicas of images from the training dataset are produced using image manipulation techniques like zooms, flips, shifts, etc. For data augmentation in testing time, it entails making numerous enhanced copies of each picture in the test set, having the model estimate for each, and then delivering an ensemble of those predictions. In this paper, both training and testing data are augmented to boost performance. The augmentation operations and scale jitter are set based on [13].

4.2. Evaluation metrics

To evaluate our method comprehensively, we assess the performance of both instance mask prediction and BBOX detection in underwater environments. We report the standard COCO metrics including Average Precision (AP), AP_{50}, and AP_{75}. AP is measured by averaging over different thresholds of IoU from 0.5 to 0.95 with a step size of 0.05. That is, AP is calculated by averaging the APs over all object categories, which is six categories in our case and all 10 IoU thresholds from 0.5 to 0.95 with a step size of 0.05. Such averaging over IoUs and categories provides a thorough evaluation rewards models with better performance. The definition of AP is given by

$$AP = \frac{1}{10} \sum_{i \in [0.5:0.95]} AP_i$$
$$AP_i = \frac{1}{|N_c|} \sum_{c \in N_c} \frac{|TP_c|}{|FP_c| + |TP_c|} \quad (12)$$

where N_c is the number of classes. TP_c and FP_c are true positives and false positives of class c.

4.3. Implementation details

The implementation of our method is based on PyTorch, which is a deep learning framework. For the backbone network, the pre-trained ResNet-101 [13] is chosen due to its competent performance [13]. Stochastic Gradient Descent (SGD) is used to optimise instance segmentation networks. The threshold is set to 0.5 for BBOX/segmentation predictions. Following the work in [13,30,31], the networks are trained for 4908 iterations. The initial Learning Rate (LR) is set to 0.001, except CenterMask whose learning rate is set to 0.0001 since a higher value causes exploding gradients. As suggested in [37], we resize the raw input image to the resolution of 1024 × 768. For the hard configurations, we run all experiments on a PC with CPU: Intel 2.60 GHz i7-10750H, GPU: GeForce RTX 2060, and RAM: 16 GB.

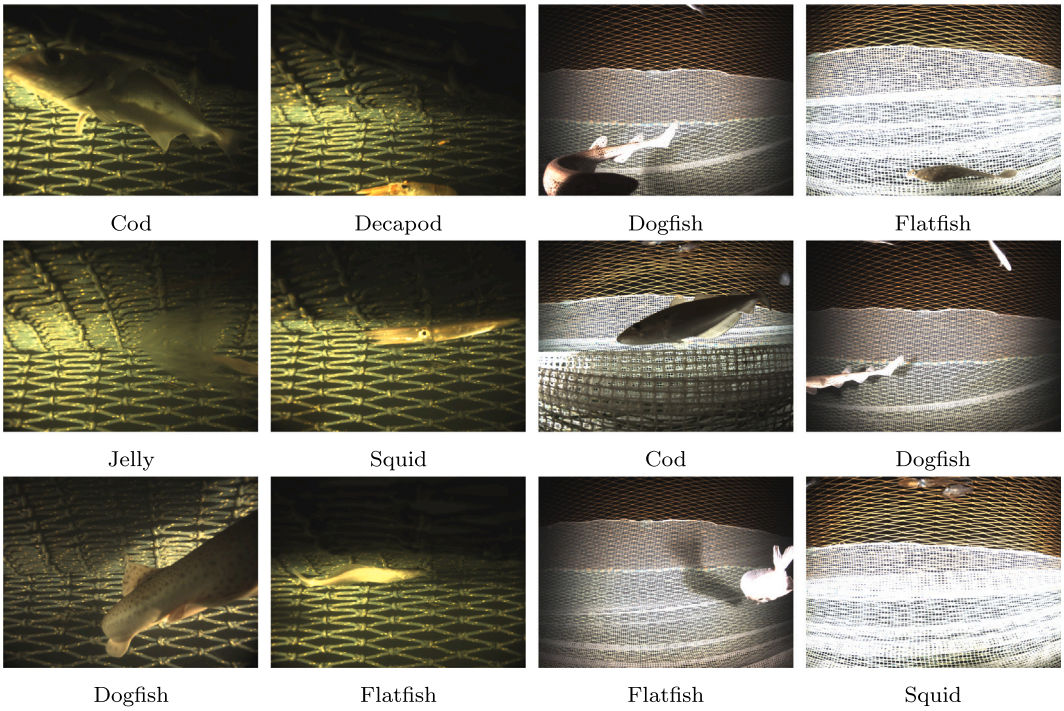

Fig. 2. The samples of source domain and target domain underwater images, including Cod, Decapod, Dogfish, Flatfish, Jelly, Squid. The first two columns are the examples of source images and the last two columns are the examples of target images.

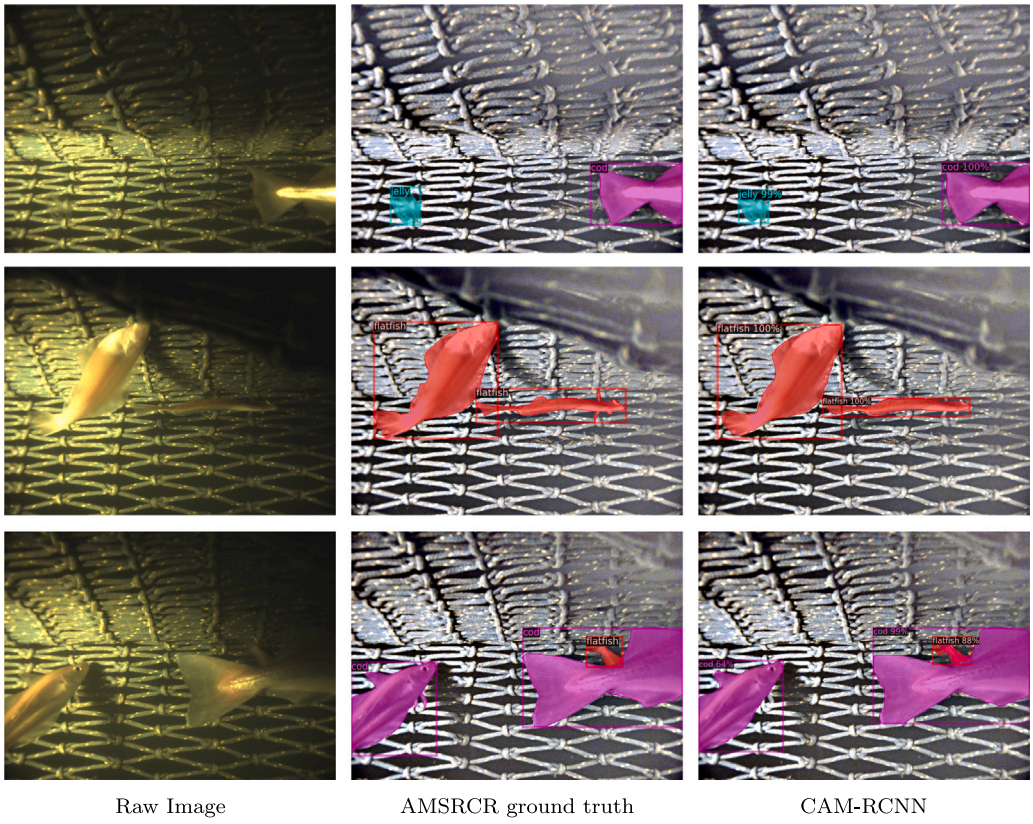

Fig. 3. Qualitative comparison in the source domain between: **(a)** raw image, **(b)** AMSRCR ground truth annotation, and **(c)** CAM-RCNN model AMSRCR prediction images respectively on the Shetland deployment test set. The prediction is performed using our best CAM-RCNN model. The first row is on source domain and second row is on target domain.

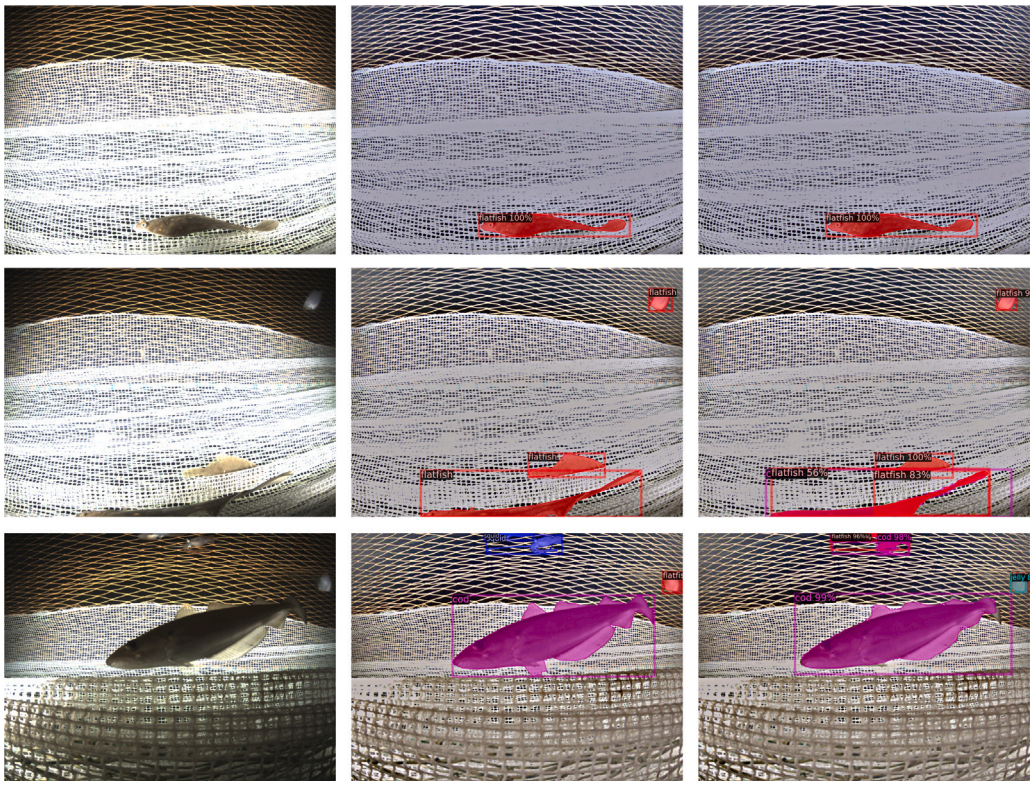

Fig. 4. Qualitative comparison in target domain between: **(a)** raw image, **(b)** AMSRCR ground truth annotation, and **(c)** CAM-RCNN model AMSRCR prediction images respectively on the Shetland deployment test set. The prediction is performed using our best CAM-RCNN model. The first row is on source domain and second row is on target domain.

4.4. Performance of instance segmentation

This section provides the instance segmentation results of evaluated in source domain and target domain aquatic datasets. We first discuss the results of source domain among different methods. Then, we explore the results of target domain for different methods.

4.4.1. Source domain comparative results

we compared our CAM-RCNN method with other methods in source domain. CAM-RCNN adopts a single CoordConv and multiple GN layers in its mask head. To deliver both mask and BBOX detections, we integrate BBOX into Matrix NMS in our method to produce instance NMS. From Table 1 and Fig. 3, there are two observations drawn:

- Our proposed method outperforms other competent methods in terms of instance mask and BBOX detection, which can achieve 39.7% of AP for instance mask and 40.2% of AP for BBOX detection.
- Our proposed method is robust for different thresholds. When setting threshold to 50 and 75 respectively, our method still provide the best results for BBOX detection, which are 56.6% and 49.1% respectively. For instance segmentation, AP_{50} can reach the best results of 57.6% and AP_{75} can provide a promising result of 44.8%.

4.4.2. Target domain comparative results

Our method significantly outperforms other advanced methods, which shows great generalisation ability in target domain. Table 2 depicts the comparative results in target domain, where we obtains two observations from it.

- With regard to the generalisation ability in target domain, CAM-RCNN provides significantly improved results both on instance mask and BBOX prediction. More specifically, the AP, AP_{50}, and AP_{75} of instance mask have the best results, which can reach 24.4%, 31.5%, and 30.2%, respectively. The AP, AP_{50}, and AP_{75} of BBOX prediction also provide the best results, which are 24.2%, 31.3%, and 27.5%, respectively.
- In terms of AP_{75}, our CAM-RCNN method can provide much better result in target domain, which can achieve the best performance of 30.2%. This demonstrates its superiority on generalisation compared to other methods.

Table 1
Quantitative Results of instance mask and BBOX of Different Methods in Source Domain. We denote BBOX AP as AP^{bb}.

Model	AP	AP_{50}	AP_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}
CenterMask [30]	33.1	46.2	41.4	34.2	46.0	39.1
CondInst [31]	28.7	38.0	35.2	28.5	38.0	33.3
MRCNN [13]	37.8	52.6	47.3	38.4	52.2	45.0
SOLOv2 [32]	39.2	51.6	**47.7**	–	–	–
Cascade RCNN [38]	16.7	36.8	14.4	18.0	38.6	14.9
YOLACT [39]	15.8	29.5	14.0	19.4	39.4	19.9
POINT-REND [40]	34.3	58.0	35.4	28.2	54.9	24.2
INSTABOOST [41]	21.3	35.3	20.0	16.9	35.2	13.2
BOXINST [42]	3.5	17.0	0.2	46.3	65.4	48.6
CAM-RCNN	**39.7**	**57.6**	44.8	**40.2**	**56.6**	**49.1**

4.4.3. Wilcoxon signed-rank test

Following [43,44], we employed the Wilcoxon signed-rank test to assess the presence of performance disparities between our proposed method and other state-of-the-art techniques. The results, outlined in Table 3, demonstrate that our method exhibits the highest mean

Table 2
Quantitative Results of instance mask and BBOX of Different Methods in Target Domain. We denote BBOX AP as AP^{bb}.

Model	AP	AP_{50}	AP_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}
CenterMask	14.9	19.8	18.2	15.9	19.6	18.8
CondInst	8.8	11.2	10.9	8.9	11.1	10.8
MRCNN	18.5	24.9	22.6	18.8	24.4	22.8
SOLOv2	14.5	19.8	16.7	–	–	–
Cascade MRCNN [38]	4.9	11.1	3.0	5.0	11.2	3.3
YOLACT [39]	8.0	18.5	4.2	7.5	18.1	3.7
POINT-REND [40]	14.3	25.7	13.7	9.6	23.2	6.0
INSTABOOST [41]	8.0	15.6	7.9	6.3	14.9	3.6
BOXINST [42]	5.1	16.5	2.0	20.0	26.4	24.2
CAM-RCNN	**24.4**	**31.5**	**30.2**	**24.2**	**31.3**	**27.5**

Table 3
Wilcoxon Signed-Rank Test Between Our CAM-RCNN and other State-Of-The-Art Method.

Model	mAP	mAP^{bb}	*p*-value
CAM-RCNN	32.1	32.2	–
CAM-RCNN vs. CenterMask [30]	24.0	25.1	4.8828e−04
CAM-RCNN vs. CondInst [31]	18.8	18.7	4.8828e−04
CAM-RCNN vs. MRCNN [13]	28.2	28.6	0.0024
CAM-RCNN vs. Cascade RCNN [38]	10.8	11.5	4.8828e−04
CAM-RCNN vs. YOLACT [39]	11.9	13.5	4.8828e−04
CAM-RCNN vs. POINT-REND [40]	24.3	18.9	9.7656e−04
CAM-RCNN vs. INSTABOOST [41]	14.7	11.6	4.8828e−04
CAM-RCNN vs. BOXINST [42]	4.3	33.2	0.0269

Table 4
The results of AP, training time, and inference speed for various methods. We denote frame per second as FPS.

Model	AP	Training time (s)	Inference speed (FPS)
CenterMask [30]	14.9	3072	4.3
CondInst [31]	8.8	2925	**4.4**
MRCNN [13]	18.5	**2706**	4.1
SOLOv2 [32]	14.5	3106	3.6
CAM-RCNN	**24.4**	**2754**	**4.1**

Average Precision (AP) and AP^{bb} values in both source and target domains. Additionally, the statistical significance, with *p*-values consistently below 0.05, indicates that the median performance difference between our CAM-RCNN and the comparative state-of-the-art methods is confidently non-zero. Our CAM-RCNN consistently outperforms other methods in terms of mean mAP and mAP^{bb} across both source and target domains. This superior performance is attributed to our designed CAGN module, embedded within the mask generation branch, which adeptly captures crucial location information for precise segmentation. Furthermore, the compounded DCE loss function enhances the method's segmentation accuracy. The generalisation capacity is further strengthened by AMSRCR and inference augmentation techniques.

4.4.4. Trade-off on training time, inference speed and AP

According to the experimental results, our method shows the best trade-off among training time, inference speed, and AP. More specifically, the following observations can be drawn from Table 4.

- In terms of the trade-off on AP, training time, and inference speed, our method significantly outperform other methods, which can 24.4% of AP with training time for 2754 s and the inference speed for 4.3 frame per second (FPS) on our laptop-based GPU.

Table 5
The ablation study on various components of our method for instance mask and BBOX prediction in Source Domain. We denote BBOX AP as AP^{bb} and AMSRCR as AMS.

Various Components				Source Domain					
CAGN	DCE	AMS.	IA	AP	AP_{50}	AP_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}
✓				36.5	51.4	45.7	37.0	50.5	42.6
✓	✓			39.1	56.2	49.1	40.0	55.7	48.4
✓	✓	✓		37.0	52.0	46.2	37.4	51.5	43.5
✓	✓	✓	✓	**39.7**	**57.6**	44.8	**40.2**	**56.6**	**49.1**

Table 6
The ablation study on various components of our method for instance mask and BBOX prediction in Target Domain. We denote BBOX AP as AP^{bb} and AMSRCR as AMS.

Various Components				Target Domain					
CAGN	DCE	AMS.	IA	AP	AP_{50}	AP_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}
✓				18.7	25.5	23.2	19.0	25.0	22.4
✓	✓			19.4	25.8	24.1	20.2	25.3	24.5
✓	✓	✓		21.0	27.7	25.5	21.1	27.2	23.9
✓	✓	✓	✓	**24.4**	**31.5**	**30.2**	**24.2**	**31.3**	**27.5**

- Here, we focus on AP, training time, and inference speed individually. In terms of AP, we can find that our method achieves the best results. In terms of training time, MRCNN provides minimum training time, which is 2706 s and the training time of our method is only sightly higher which is 2754 s. In terms of inference speed, CondInst has the fastest inference speed while it has much poor results on AP and training time, which are 8.8% and 2925 s. In contrast, our method is only sightly slower than CondInst which only reduce 4.4 to 4.1 of FPS but the AP of our method can achieve 24.4% and the training time is only required for 2754 s.

4.4.5. Ablation study

To identify the contributions of various components in our method, extensive experiments of ablation study are carried out to determine their functions in our proposed model. The performance of segmentation and detection is provided in Tables 5 and 6, where the improvement is shown by adding one more components at each stage. Tables 5 and 6 demonstrate the performance improvements of source domain and target domain respectively, where coordinate-awareness with group normalisation is denoted as CAGN, Dice binary cross entropy loss is denoted as DCE, Automated Multi-Scale Retinex with Color Restoration is denoted as AMS., inference augmentation is denoted as IA.

For source domain, we can see that the performance of AP and AP^{bb} are 36.5% and 37.0% when we introduce CAGN. It will bring the performance gain as 2.6% if we introduce DCE. When AMS and IA are brought into instance segmentation, it will further improve performance 39.7% of AP and 40.2% of AP^{bb}.

For target domain, it concentrates on generalisation ability to evaluate the performance on the data which are significantly important from previous seen data. The generalisation performance of AP and AP^{bb} are 18.7% and 19.0%. If we introduce DCE, AMS, and IA into instance segmentation, the performance gain of them are 0.7%, 1.6%, and 3.4% of AP for instance mask. For the BBOX detection, the performance gains of them are brought as 1.2%, 0.9%, and 3.1% of AP^{bb}.

5. Conclusion

This paper proposes a novel coordinate-aware instance segmentation method to detect and segment aquatic animals. Our proposed CAM-RCNN method fully utilises the strengths of CoordConv and Group Normalisation to ensure generalise well in various scenarios. Moreover, we propose a compound dice and cross-entropy loss to further boost prediction performance. In addition, automated multi-scale

retinex with colour restoration approach is also used to carry out image enhancement to further boost the generalisation ability. Furthermore, inference augmentation is employed with instance NMS to further enhance the robustness and performance.

To evaluate the fish detection and species identification performance of our method in deep sea scenarios, we collect two deep sea datasets from North Sea in 2019 and 2022, as source domain dataset and target domain dataset. An extensive comparison was conducted on a number of advanced instance segmentation methods to identify the superiority of accuracy and generalisation ability for our method. Experimental results show that our CAM-RCNN method had a noticeable improvement in terms of generalisation ability and accuracy. Moreover, an ablation study was carried out to investigate the contributions of various components in our method. As a result, our CAM-RCNN method obtained the best performance on both source domain and target domain evaluation, where AP of instance mask and BBOX prediction were 39.7% and 40.2% in the source domain; 24.4% and 24.2% in the target domain, which were significantly better than other competent methods.

In the future, we aim to cover a much broader spectrum of aquatic animal classes along with pixel-level and instance-level annotations by experts. We will also ensure that the data distribution across collected data and instance classes will be more balanced. In addition, we will develop a lightweight version of our method to trade off the training and inference speeds so that our method can be deployed in low-cost devices and running in real time. Ultimately, our system will be used in real time to relay information to a gate system in fishing trawls, allowing for the in-situ release, alive and well, of unwanted catches. This is turn with contribute to the elimination of the problem of discards and bycatch and improve the sustainability of fishing operations to enhance food security and maintain biodiversity.

CRediT authorship contribution statement

Dewei Yi: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Hasan Bayarov Ahmedov:** Conceptualization, Data curation, Methodology, Software, Validation, Visualization, Writing – original draft. **Shouyong Jiang:** Conceptualization, Formal analysis, Methodology, Supervision, Writing – review & editing. **Yiren Li:** Data curation. **Sean Joseph Flinn:** Data curation. **Paul G. Fernandes:** Conceptualization, Funding acquisition, Investigation, Project administration, Resources, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

Acknowledgements

This work was supported in part by the Fisheries Innovation & Sustainability and U.K. Department for Environment, Food & Rural Affairs under Grant numbers: FIS039 and FIS045A.

References

[1] P.G. Fernandes, R.M. Cook, Reversal of fish stock decline in the northeast atlantic, Curr. Biol. 23 (15) (2013) 1432–1437.
[2] R. Garcia, R. Prados, J. Quintana, A. Tempelaar, N. Gracias, S. Rosen, H. Vågstøl, K. Løvall, Automatic segmentation of fish using deep learning with application to fish size measurement, ICES J. Mar. Sci. 77 (4) (2020) 1354–1366.
[3] P.G. Fernandes, K. Coull, C. Davis, P. Clark, R. Catarino, N. Bailey, R. Fryer, A. Pout, Observations of discards in the scottish mixed demersal trawl fishery, ICES J. Mar. Sci. 68 (8) (2011) 1734–1742.
[4] I.C. Avila, K. Kaschner, C.F. Dormann, Current global risks to marine mammals: taking stock of the threats, Biol. Cons. 221 (2018) 44–58.
[5] S. Oliver, M. Braccini, S.J. Newman, E.S. Harvey, Global patterns in the bycatch of sharks and rays, Mar. Policy 54 (2015) 86–97.
[6] J. Simmonds, D.N. MacLennan, Fisheries Acoustics: Theory and Practice, John Wiley & Sons, 2008.
[7] Y. Liu, D. Zhang, Q. Zhang, J. Han, Part-object relational visual saliency, IEEE Trans. Pattern Anal. Mach. Intell. 44 (7) (2021) 3688–3704.
[8] C. Yan, T. Teng, Y. Liu, Y. Zhang, H. Wang, X. Ji, Precise no-reference image quality evaluation based on distortion identification, ACM Trans. Multimed. Comput., Commun. Appl. (TOMM) 17 (3s) (2021) 1–21.
[9] Z. Shao, J. Han, K. Debattista, Y. Pang, Textual context-aware dense captioning with diverse words, IEEE Trans. Multimed. (2023).
[10] C. Fu, R. Liu, X. Fan, P. Chen, H. Fu, W. Yuan, M. Zhu, Z. Luo, Rethinking general underwater object detection: Datasets, challenges, and solutions, Neurocomputing 517 (2023) 243–256.
[11] M. Gao, F. Zheng, J.J. Yu, C. Shan, G. Ding, J. Han, Deep learning for video object segmentation: A review, Artif. Intell. Rev. 56 (1) (2023) 457–531.
[12] H. Huang, H. Zhou, X. Yang, L. Zhang, L. Qi, A.-Y. Zang, Faster R-CNN for marine organisms detection and recognition using data augmentation, Neurocomputing 337 (2019) 372–384.
[13] K. He, G. Gkioxari, P. Dollár, R. Girshick, Mask r-cnn, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2961–2969.
[14] H. Qin, X. Li, J. Liang, Y. Peng, C. Zhang, DeepFish: Accurate underwater live fish recognition with a deep architecture, Neurocomputing 187 (2016) 49–58.
[15] C. Yan, Z. Li, Y. Zhang, Y. Liu, X. Ji, Y. Zhang, Depth image denoising using nuclear norm and learning graph model, ACM Trans. Multimed. Comput., Commun. Appl. (TOMM) 16 (4) (2020) 1–17.
[16] C. Yan, B. Gong, Y. Wei, Y. Gao, Deep multi-view enhancement hashing for image retrieval, IEEE Trans. Pattern Anal. Mach. Intell. 43 (4) (2020) 1445–1451.
[17] Y. Liu, M. Duanmu, Z. Huo, H. Qi, Z. Chen, L. Li, Q. Zhang, Exploring multi-scale deformable context and channel-wise attention for salient object detection, Neurocomputing 428 (2021) 92–103.
[18] C. Yan, Y. Hao, L. Li, J. Yin, A. Liu, Z. Mao, Z. Chen, X. Gao, Task-adaptive attention for image captioning, IEEE Trans. Circuits Syst. Video Technol. 32 (1) (2021) 43–51.
[19] Z. Shao, J. Han, D. Marnerides, K. Debattista, Region-object relation-aware dense captioning via transformer, IEEE Trans. Neural Netw. Learn. Syst. (2022).
[20] R. Garcia, R. Prados, J. Quintana, A. Tempelaar, N. Gracias, S. Rosen, H. Vågstøl, K. Løvall, Automatic segmentation of fish using deep learning with application to fish size measurement, ICES J. Mar. Sci. 77 (4) (2020) 1354–1366.
[21] D.A. Konovalov, A. Saleh, M. Bradley, M. Sankupellay, S. Marini, M. Sheaves, Underwater fish detection with weak multi-domain supervision, in: 2019 International Joint Conference on Neural Networks, IJCNN, IEEE, 2019, pp. 1–8.
[22] A. Salman, A. Jalal, F. Shafait, A. Mian, M. Shortis, J. Seager, E. Harvey, Fish species classification in unconstrained underwater environments based on deep learning, Limnol. Oceanogr.: Methods 14 (9) (2016) 570–585.
[23] A. Salman, S.A. Siddiqui, F. Shafait, A. Mian, M.R. Shortis, K. Khurshid, A. Ulges, U. Schwanecke, Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system, ICES J. Mar. Sci. 77 (4) (2020) 1295–1307.
[24] C.O. Ancuti, C. Ancuti, C. De Vleeschouwer, P. Bekaert, Color balance and fusion for underwater image enhancement, IEEE Trans. Image Process. 27 (1) (2017) 379–393.
[25] S. Parthasarathy, P. Sankaran, An automated multi scale retinex with color restoration for image enhancement, in: National Conference on Communications, IEEE, 2012, pp. 1–5.
[26] S. Mittal, S. Srivastava, J.P. Jayanth, A survey of deep learning techniques for underwater image classification, IEEE Trans. Neural Netw. Learn. Syst. (2022).
[27] L. Li, B. Dong, E. Rigall, T. Zhou, J. Dong, G. Chen, Marine animal segmentation, IEEE Trans. Circuits Syst. Video Technol. 32 (4) (2021) 2303–2314.
[28] H.S. Demir, J.B. Christen, S. Ozev, Energy-efficient image recognition system for marine life, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. 39 (11) (2020) 3458–3466.
[29] C.-H. Yeh, C.-H. Lin, L.-W. Kang, C.-H. Huang, M.-H. Lin, C.-Y. Chang, C.-C. Wang, Lightweight deep neural network for joint learning of underwater object detection and color conversion, IEEE Trans. Neural Netw. Learn. Syst. (2021).
[30] Y. Lee, J. Park, Centermask: Real-time anchor-free instance segmentation, in: Proc. IEEE Conf. Comp. Vis. Patt. Recogn., 2020, pp. 13906–13915.

[31] Z. Tian, C. Shen, H. Chen, Conditional convolutions for instance segmentation, in: Proc. Eur. Conf. Comput. Vis., Springer, 2020, pp. 282–298.
[32] X. Wang, R. Zhang, T. Kong, L. Li, C. Shen, Solov2: Dynamic and fast instance segmentation, in: Advances in Neural Inf. Process. Syst., vol. 33, 2020, pp. 17721–17732.
[33] R. Liu, J. Lehman, P. Molino, F. Petroski Such, E. Frank, A. Sergeev, J. Yosinski, An intriguing failing of convolutional neural networks and the coordconv solution, in: Advances in Neural Inf. Process. Syst., vol. 31, 2018.
[34] M.A. Islam, S. Jia, N.D. Bruce, How much position information do convolutional neural networks encode?, 2020, arXiv preprint arXiv:2001.08248.
[35] X. Wang, T. Kong, C. Shen, Y. Jiang, L. Li, Solo: Segmenting objects by locations, in: Proc. Eur. Conf. Comput. Vis., Springer, 2020, pp. 649–665.
[36] Y. Wu, K. He, Group normalization, in: Proc. Eur. Conf. Comput. Vis., 2018, pp. 3–19.
[37] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: IEEE Conf. Comp. Vis. Patt. Recogn., 2017, pp. 2117–2125.
[38] Z. Cai, N. Vasconcelos, Cascade R-CNN: High quality object detection and instance segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 43 (5) (2019) 1483–1498.
[39] D. Bolya, C. Zhou, F. Xiao, Y.J. Lee, Yolact: Real-time instance segmentation, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 9157–9166.
[40] A. Kirillov, Y. Wu, K. He, R. Girshick, Pointrend: Image segmentation as rendering, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9799–9808.
[41] H.-S. Fang, J. Sun, R. Wang, M. Gou, Y.-L. Li, C. Lu, Instaboost: Boosting instance segmentation via probability map guided copy-pasting, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 682–691.
[42] Z. Tian, C. Shen, X. Wang, H. Chen, Boxinst: High-performance instance segmentation with box annotations, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 5443–5452.
[43] D. Li, M.-R. Jiang, M.-W. Li, W.-C. Hong, R.-Z. Xu, A floating offshore platform motion forecasting approach based on EEMD hybrid ConvLSTM and chaotic quantum ALO, Appl. Soft Comput. (2023) 110487.
[44] H. Mei, Y. Liu, Z. Wei, D. Zhou, X. Wei, Q. Zhang, X. Yang, Exploring dense context for salient object detection, IEEE Trans. Circuits Syst. Video Technol. 32 (3) (2021) 1378–1389.

Dewei Yi received his Ph.D. degree from Loughborough University, U.K, in 2018. He is currently working in the School of Natural and Computing Sciences at University of Aberdeen. His current research interests include applied machine learning, personalised systems, AI in healthcare, hybrid intelligent systems, intelligent vehicles, vehicular network, and precise agriculture.

Hasan Bayarov Ahmedov received an MEng degree in Computing Science at University of Aberdeen, UK. His research interests include object detection and deep learning.

Shouyong Jiang received the B.Sc. degree in information and computation science and the M.Sc. degree in control theory and control engineering from Northeastern University, Shenyang, China, in 2011 and 2013, respectively, and the Ph.D. degree in computer science from De Montfort University, Leicester, U.K., in 2017. His current research interests include AI optimisation, evolutionary computation, and machine learning.

Yiren Li is currently pursuing an Ph.D. degree in Computing Science at University of Aberdeen, UK. Her research interests include image processing and deep learning.

Sean Joseph Flinn received an M.S. degree in Marine conservation at University of Aberdeen, UK. His research interests include marine conservation and species monitoring .

Paul G. Fernandes received his B.Sc. and Ph.D. in Marine Biology from the University of Liverpool's Port Erin Marine Laboratory. He is now a professor in the Heriot-Watt University and appointed as Bicentennial Research Leader. Before joining Heriot-Watt, he was at the University of Aberdeen for 11 years and, prior to that, he worked at the Marine Laboratory Aberdeen (now Marine Scotland Science) for over 16 years. There he focused on fisheries surveys initially and, latterly, fish stock assessment, where he led the Sea Fisheries Group. His early career consisted of stints in Ireland, setting up their fisheries acoustics programme, and in Bolivia working on the artisanal fisheries of Lake Titicaca.