

Computer vision in aquaculture: a case study of juvenile fish counting

Krishna Moorthy Babu, Daniel Bentall, David T. Ashton, Morgan Puklowski, Warren Fantham, Harris T. Lin, Nicholas P. L. Tuckey, Maren Wellenreuther & Linley K. Jesson

To cite this article: Krishna Moorthy Babu, Daniel Bentall, David T. Ashton, Morgan Puklowski, Warren Fantham, Harris T. Lin, Nicholas P. L. Tuckey, Maren Wellenreuther & Linley K. Jesson (2022): Computer vision in aquaculture: a case study of juvenile fish counting, Journal of the Royal Society of New Zealand, DOI: [10.1080/03036758.2022.2101484](https://doi.org/10.1080/03036758.2022.2101484)

To link to this article: <https://doi.org/10.1080/03036758.2022.2101484>



Published online: 03 Aug 2022.



Submit your article to this journal [↗](#)



View related articles [↗](#)



View Crossmark data [↗](#)

RESEARCH ARTICLE



Computer vision in aquaculture: a case study of juvenile fish counting

Krishna Moorthy Babu ^a, Daniel Bentall ^a, David T. Ashton ^b,
Morgan Puklowski ^b, Warren Fantham ^b, Harris T. Lin ^a,
Nicholas P. L. Tuckey ^b, Maren Wellenreuther ^{b,c} and Linley K. Jesson ^a

^aThe New Zealand Institute for Plant and Food Research Limited, Lincoln, New Zealand; ^bThe New Zealand Institute for Plant and Food Research Limited, Nelson, New Zealand; ^cThe University of Auckland, Auckland, New Zealand

ABSTRACT

In aquaculture breeding or production programmes, counting juvenile fish represents a considerable cost in terms of the human hours needed. In this study, we explored the use of two state-of-the-art machine learning architectures (Single Shot Detection, hereafter SSD and Faster Regions with convolutional neural networks, hereafter Faster R-CNN) to augment a manual image-based juvenile fish counting method for the Australasian snapper (*Chrysophrys auratus*) bred at The New Zealand Institute for Plant and Food Research Limited. We tested model accuracy after tuning for confidence thresholds and non-maximal suppression overlap parameters, and implementing a bias correction using a Poisson regression model. Validation of image data showed that after tuning, bias-corrected SSD and Faster R-CNN models had mean absolute percent errors (MAPE) of less than 10%, with SSD having MAPE of less than 5%. Comparison of the results with those from manual counts showed that, while manual counts are slightly more accurate (MAPE = 1.56), the machine learning methods allow for more rapid assessment of counts and thus facilitating a higher throughput. This work represents a first step for deploying machine learning applications to an existing real-life aquaculture scenario and provides a useful starting point for further developments, such as real-time counting of fish or collecting additional phenotypic data from the source images.

ARTICLE HISTORY

Received 23 December 2021
Accepted 11 July 2022


KEYWORDS

Computer vision; object detection; *Chrysophrys auratus*; aquaculture; imaging

Introduction

Aquaculture has a central role to play in feeding the world (FAO 2018). Continued growth in human population sizes and a focus on high quality animal protein is predicted to increase this need even more in the future (Godfray et al. 2010; Béné et al. 2015). To meet this challenge, aquaculture researchers and producers are seeking innovative ways to make aquaculture production more efficient and to improve fish quality and welfare (Antonucci and Costa 2020). One area that has seen considerable focus is the use of

CONTACT David Ashton  David.Ashton@plantandfood.co.nz

 Supplemental data for this article can be accessed online at <https://doi.org/10.1080/03036758.2022.2101484>.

© 2022 The New Zealand Institute for Plant and Food Research Limited

new technologies for counting and measuring fish in aquaculture facilities (Fu and Yuna 2022; Rasmussen et al. 2022), and similar technologies are also being developed for informing fisheries assessments and for monitoring wild fish populations (Connolly et al. 2021). Fish counting is critically important in aquaculture breeding and production programmes for operational reasons (e.g. moving fish), identifying and mitigating stressors (e.g. evaluating fish density), and to optimise the conditions for fish health and growth (e.g. estimating required feed amount). For aquaculture breeding programmes this information is also used to inform the selection of the best performing individuals which continue to the next generation as broodstock (Gjedrem and Robinson 2014; Ashton et al. 2019; Valenza-Troubat et al. 2022).

Innovative applications that seek to measure and/or count fish are challenging since fish are sensitive, easily stressed, and free to move in an environment in which lighting, visibility, and stability are generally not controllable, and the hardware must operate underwater or in wet locations (Mathiassen et al. 2011). However, counting and measuring fish represents a considerable operational cost in terms of the manual hours needed, particularly for juvenile fish, which can number in the tens to hundreds of thousands per tank (Li et al. 2020). Manual counts also increase the potential for intra- and inter-observer variability, potentially leading to increased measurement errors. Consequently, automated fish counting methods have been a focus for the industry for some time, and various approaches have been trialled including potentiometric bridges (e.g. SR-1601, Smith-Root, WA, USA) where fish are pumped from tank to tank and, more recently, a variety of computer vision approaches (Li et al. 2021). Examples of these include the commercial range manufactured by Vaki (Kopavogur, Iceland) and systems such as XperCount focusing specifically on shrimp (xpertSea, QC, Canada). In recent years, the application of machine learning or deep learning in computer vision allows for automated object recognition and detection ability (Li et al. 2021). Thus, generalised automation of this task using image-based analyses represents a significant opportunity in this area, and machine-learning models provide an important toolset that enables automation, with the added potential to increase throughput and, with improved models, accuracy (Fu and Yuna 2022).

Computer vision based object detection can be accomplished through a variety of methods (Yang et al. 2021). Prior to deep learning, object detection was a multi-step process, beginning with edge detection and extraction of features using techniques such as Scale Invariant Feature Transform (SIFT) (Gupta et al. 2019), Speeded Up Robust Feature (SURF) (Wang et al. 2019), Histogram of Oriented Gradients (HOG) or Haar Cascades (Zhang et al. 2021), then images were compared with existing object models, typically at multi-scale sizes, to detect and locate objects in the picture. More recently, deep learning methods, like Convolutional Neural Networks (CNN) or versions of the You Only Look Once (YOLO) algorithm and its subsequent advanced versions, have been shown to extract complicated features with high prediction accuracy (Yang et al. 2021; Mahanty et al. 2022).

A particular challenge in computer vision and deep learning is the detection of small objects. Detection of small objects can be particularly problematic as the objects take up a smaller amount of the image and therefore contain less information content for machine learning models to base predictions on (Bochkovskiy et al. 2020). Several techniques exist to improve the detection of small objects such as increasing the image capture resolution,

increasing the model's input resolution, tiling images, generating more data via augmentation, auto learning model anchors and filtering out extraneous classes (Solawetz 2020). However, these techniques can also increase the amount of information that models have available which correlate with the size of models and the associated computation time. Most state-of-the-art detectors struggle with small object detection, particularly so if there is size heterogeneity in the image (Ge et al. 2020; Nguyen et al. 2020).

Plant & Food Research has been running selective breeding programmes for the native finfish species Australasian snapper (*Chrysophrys auratus*) and silver trevally (*Pseudocaranx georgianus*) since 2016 at its Finfish Facility in Nelson, New Zealand (Ashton et al. 2019; Baesjou and Wellenreuther 2021; Valenza-Troubat et al. 2022). As part of the on-growing procedures, fish are routinely moved between tanks to avoid overcrowding. This process is conducted by anaesthetising fish, collecting them in plastic bins partially filled with seawater and then transporting them to a new tank. A digital image is taken of each bin to count fish after the tank move is completed (see Figure 1). During this process, an initial live estimate of fish numbers can be produced and an accurate count is subsequently produced manually from images after the tank move.

In the present work, we explored the use of two state-of-the-art model architectures to automate the counting process of juvenile snapper in the hatchery. The first model was a Faster Regions with convolutional neural networks model (R-CNN), while the second model applied a Single Shot Detection (SSD) approach. Faster R-CNN is a two-stage method that first generates region proposals, and then targets the boundary boxes and category predictions of the region proposals. In contrast, SSD is a one-stage detection method that does not need to select region proposals, but use regression to directly predict the positioning of boxes and object categories, which further reduces the running time.

We first outline the steps used to implement the models and then we compare the results from the models with those generated using manual methods. Finally, we



Figure 1. A, Sample image of SSD (Single shot multibox object detection) model prediction on juvenile fish bins. The fish sizes in the 394 images used in this study ranged from 20 to 80 mm and the number of fish instances present in the images ranged from 12 to 618. **B,** Sample image of Faster R-CNN (Regions with convolutional neural networks) model prediction on juvenile fish bins. The fish sizes in the 394 images used in this study ranged from 20 to 80 mm and the number of fish instances present in the images ranged from 12 to 618.

discuss the future directions of this work, and highlight areas that are particularly fruitful to assist aquaculture operations.

Materials and methods

Data

We assessed the ability of computer vision models to detect small fish using image data of juvenile snapper in the breeding programme at the Plant and Food Research's Nelson Research Centre Finfish Facility in New Zealand. We compared this to manual counts as usually conducted as part of the fish processing pipelines at Plant and Food and referenced these both against a 'true' count taken from annotated images.

The dataset consists of images of snapper at different age and size classes taken with a Canon Powershot camera. Fish age groups ranged from 26 to 97 days old post hatch grouped in 3- to 7-day periods. The number of fish instances present in any given image ranged from 12 to 618. The size classes of individual juvenile snapper in the images ranged from 20 to 80 mm.

Fish were sampled from tanks into fish bins and images were taken for each bin. Manual counts of images were performed while visually inspecting the images by an observer on the PC, and then individually identified fish were marked with image processing software (Microsoft Paint or Nikon NIS-Elements) by clicking on each fish separately, leaving a mark on the counted fish. In this way, the visual signature reduces the chance of recounting the same fish. The total number of dots (i.e. counted fish) was then summed up per image. In total, 394 images (in JPG format) were counted and used. For the model training, each fish in the fish bin images was labelled manually using CVAT (Computer Vision Annotation Tool; Sekachev et al. 2020). Each image had thus associated annotations (in XML format), consisting of the filename, image height, image width, and bounding box coordinates of all the fishes present in a particular image.

Counts estimated either by computer vision or manually were compared to a 'true count' of the number of bounding boxes labelled during model training. The manual bounding box labels were considered more accurate than manual counts as observers spent considerable effort in annotating each fish in the image.

Model training

Model training and selection constituted an iterative process by which the best method for object detection was first selected, and then the best method for counting fish was evaluated by testing combinations of parameter values and thresholds as well as evaluating statistical methods for correcting for bias.

We derived unbiased model performance estimates by training and testing the models using the k -fold cross-validation approach. In this method, the dataset is randomly split into k equal-sized folds, and k models are trained, each with $k-1$ folds used for training, and one fold for testing, such that each model has a different combination of folds for training and a different test fold. We calculated the performance of each model on its test data, resulting in a final average and standard deviation of each metric. Here, we

chose k to be 3 to reduce computation time. This resulted in the testing datasets containing one-third of the dataset (131 to 132 images each). The computer-vision model training processes further split their training folds to allow for early stopping using 10% of the data (26 images).

For object detection we selected one state-of-the-art method from each of the two main branches of deep-learning-based object detection: the two-stage method Faster R-CNN (Ren et al. 2015) and the one-stage method Single Shot Multibox (Liu et al. 2016). As a generalisation, two-stage detectors have higher localisation and object recognition accuracy, whereas one-stage detectors achieve higher inference speeds. The first stage of Faster R-CNN (Region Proposal Network) proposes candidate object bounding boxes. These are then extracted from each candidate region in the second stage using a RoI Pooling operation for subsequent classification and bounding-box regression tasks. In SSD, boxes are predicted from input images directly without the region proposal step, thus they are time efficient and more suitable for real-time devices.

The model parameters were initialised from models pre-trained on the ImageNet (Russakovsky et al. 2015) and COCO (Lin et al. 2014) datasets, then trained with the juvenile fish bins dataset. This technique is known as transfer learning, and it allows the training of deep machine learning models with small datasets. The models were end-to-end trained on a high performance cluster computing environment equipped with 4 units of 16 GB Nvidia Tesla V100 GPU and 4 units of 40 GB Nvidia A100 GPU.

Data augmentation is a technique that randomly perturbs data and labels during training to increase data diversity and potentially increase model performance and generalisability. We used online data augmentation, meaning that the augmentations were applied randomly to the images at each training iteration. During SSD model training, we randomly resized images in the relative range 0.4 to 1.0, preserving the aspect ratio of the image, then randomly cropped to 512×512 pixels. During Faster R-CNN model training, we randomly cropped images to 2000×2000 pixels. Additionally, during training with both models, random horizontal flipping was applied.

The objects in the image can be of different sizes and shapes, and to allow for this, the model predicts multiple bounding boxes of different sizes and aspect ratios at each location in the image. The vast majority of these boxes have very low confidence, and are filtered out. Ideally, for each instance of fish in the image, the model should return only a single bounding box; however, usually more than one box remains for each object. To select the best bounding box from the multiple predicted bounding boxes, these object detection models use non-maximum suppression (NMS). This method decides which boxes belong to the same object based on an overlap threshold, and then selects only the highest confidence box for each object. During development, we adjusted the threshold to suit the high degree of overlap that can occur between different fish.

Object detection models produce a list of bounding box predictions and associated confidence scores, including many of low confidence, but the task is to produce a count of the boxes, so it is necessary to count the boxes that have a confidence above a certain threshold. This threshold can be tuned to optimise different metrics on the training dataset. It may be desirable to balance the precision and recall of the thresholded

model. If these are given equal weight, this can be done by maximising the F1 score. Another approach is to tune it to give an unbiased count; the sum of the predicted count across the training dataset should equal the sum of the actual count. Because the counts in this study vary more than one magnitude in value, to avoid biasing the tuning towards higher-count images, we use the mean percentage error (MPE) rather than the mean error. An alternative method of removing any count bias is to train a Poisson regression model to generate ‘corrected’ counts from the ‘raw’ predicted counts, i.e. counts predicted from the confidence-thresholded model. This can also correct for bias related to the count magnitude, so can improve the performance of MPE-tuned counts. This correction model may be further improved by including other variables with a relationship to the raw predicted counts, and in this case we also included a parameter for the area of the bounding box (see Results).

Evaluation metrics

For the detection task, we adopted a common evaluation technique known as mean average precision (mAP) calculated based on the Pascal VOC 2007 challenge (Everingham et al. 2010). The average precision algorithm gives an overall view of model performance by matching predicted boxes to labelled boxes in order of prediction confidence, then averaging the precision across a range of 11 different confidence threshold values. This penalises missing object instances (false negatives), duplicate detections of one instance and false positive detections. Matches are defined as when the area of intersection between the predicted box and the labelled box, divided by the area of the union of those boxes is greater than a threshold, typically, as here, 0.5. We did not use mAP for model selection, instead it was calculated using the test folds as a means to evaluate final performance.

The mAP value averages across confidence levels; however, to use the outputs of an object detection model for counting, we must select a specific confidence threshold. The performance of a model averaged across confidence levels may not predict well the performance of the model at a specific confidence threshold. Therefore, to evaluate the confidence-thresholded models, we examined a number of metrics: the F1 score (the harmonic mean between precision and recall), precision, recall and Mean Absolute Percentage Error. These metrics were used for selecting optimal NMS overlap and confidence thresholds, so were calculated using the training folds.

To evaluate models for the fish counting task, we calculated the mean absolute percentage error (MAPE) and the explained deviance (D^2). MAPE is similar to the mean absolute error, commonly used in counting evaluation (Gao et al. 2020), but uses the percentage count error rather than the count error, which means it scales with the sample count, as the error is expected to. D^2 is a generalisation of the coefficient of determination (R^2) suitable for use with non-normal errors, such as count data, where the error is generally proportional to the count. The actual fish counts were taken from the labelled images and compared to manual fish counts taken using clickers and predicted counts from the models, allowing us to compare the model’s counting performances to manual counting performance. These metrics were the final performance metrics, so were calculated using the test folds.

Results

Manual data processing

The manual count processing was done using a point and click PC-based workflow run on the digital images captured during a tank move. Average click speeds during counting tend to range between 1 and 2 clicks per second. Fish numbers in each batch are highly variable, but can be up to 10,000 individuals. As such the amount of time required to manually count a batch of fish after a tank move can be up to 3 h (staff dependent), whereas batch processing digital images using the models presented here should return a count across 100 bins of fish in approximately 30 s.

Object detection

Figure 1 shows an example of predicted bounding boxes for juvenile fish counting. As shown in Table 1, the best model using the Faster-RCNN object detector resulted in 87.78% of average precision averaged across 3 folds with standard deviation of 0.40%. Similarly, the best model using the SSD object detector resulted in 90.77% of average precision averaged across 3 folds with a percent standard deviation of 0.4. Table 1 also shows the inference speed of both models measured during prediction; SSD achieved nearly half the time that Faster R-CNN took to predict a single image. This suggests that while both SSD and Faster R-CNN were similar in terms of average precision, SSD is faster at detecting juvenile fish instances.

However, both models struggled to detect some of the fish instances, particularly when the fish were overlapping (see Figure 2). In Figure 2, the actual image contains 209 fish instances, whereas in predicted image the SSD model was only able to predict 170 and Faster R-CNN was only able to predict 106 fish instances. The best mean average precision (91%) was achieved with an NMS of 0.6 (Figure 3). Even by adjusting the NMS threshold, the model still failed to predict fish with extremely high levels of overlap. To understand why Faster R-CNN performed poorly at higher values of NMS, the precision and recall were calculated with a fixed confidence threshold of 0.5 (Figure 4). Compared to SSD, Faster R-CNN's precision reduces and becomes more variable at higher NMS overlap thresholds, while recall is similar. This implies that Faster R-CNN produces more excess object predictions, thus tuning the NMS overlap threshold is important for removing these excess predictions while at the same time balancing the need to detect overlapping objects.

Evaluation of different NMS overlap thresholds across confidence thresholds (Figure 5) showed a maximum F1 for the Faster R-CNN architecture at an overlap threshold of 0.4. For SSD the maximum F1 was as an NMS overlap of 0.5. For many values of NMS, F1 was relatively flat for Faster R-CNN across confidence thresholds. However, the peak confidence threshold for SSD was generally 0.4 or 0.5.

Table 1. Validation result comparison between Faster R-CNN (Regions with convolutional neural networks) and SS (Single Shot Multibox Detection) at NMS 0.5.

Approach	Model	Backbone	mAP (%)	STD (%)	Inference speed (ms/image)
One-stage	SSD	Resnet50	90.77	0.4	184
Two-stage	Faster R-CNN	Resnext101-FPN	87.78	0.34	324



Figure 2. Examples where the detector struggled to accurately count all fish in an image. In this instance the actual image contains 209 fish. The predicted number of fish instances in the given image were 170 (SSD) and 106 (Faster R-CNN).

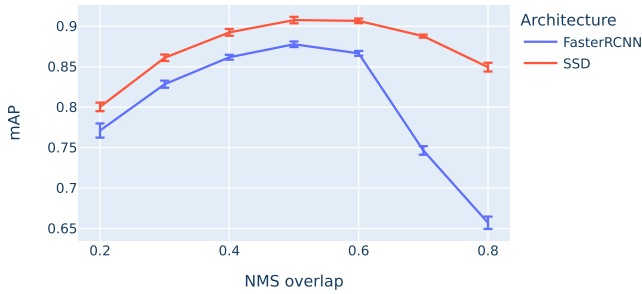


Figure 3. Non-maximum suppression (NMS) vs mean Average Precision (mAP) for the two architectures (SSD and Faster R-CNN) calculated on the test data folds. mAP was not used as a metric to optimise the NMS overlap, it is reported here as a stand-alone evaluation of final object detection performance.

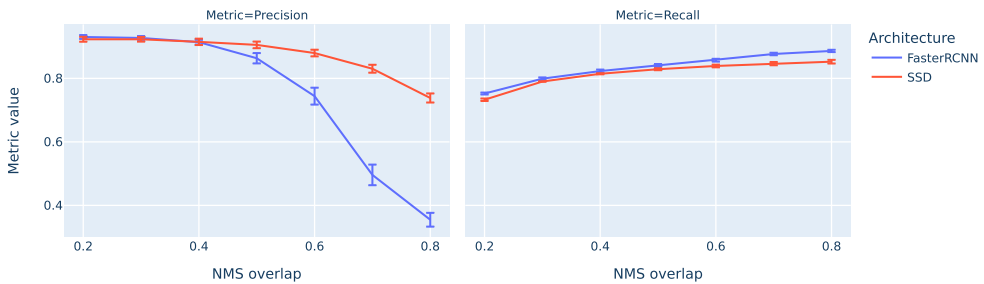


Figure 4. Precision and Recall for seven different NMS overlap thresholds using Faster R-CNN and SSD architecture at a fixed confidence threshold of 0.5. These metrics were calculated on the training data folds to enable final selection of model and tuning parameters.

Our trials of two methods for selecting the optimal confidence threshold – maximising the F1 score, and minimising the absolute value of the mean percentage error (equivalent to the mean absolute percentage error) revealed clear trade-offs between metrics across

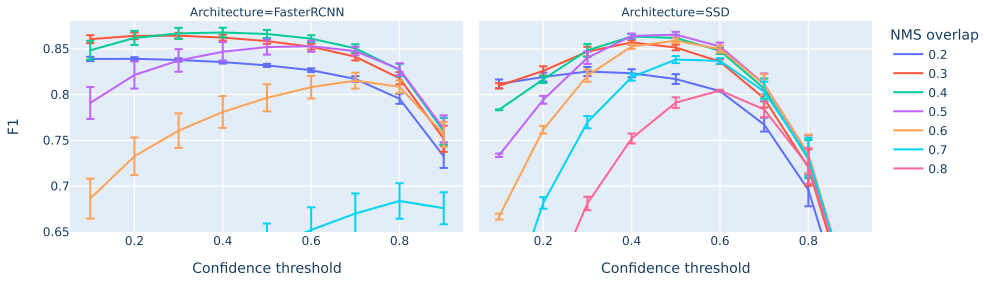


Figure 5. F1 scores across NMS overlap and confidence thresholds for Faster R-CNN and SSD architectures. F1 was calculated on the training data folds to enable final selection of model and tuning parameters.

confidence thresholds (Figure 6). For example, as precision increased, recall decreased. Low confidence thresholds tended to undercount (MPE < 0) while over-counting (MPE > 0) occurred at higher confidence thresholds. However, achieving an MPE of 0 (i.e. the predicted count matched the labelled count) was possible on the training data set. The best confidence thresholds were found to be 0.4 and 0.5 for F1-tuned Faster R-CNN and SSD, respectively, and 0.2 and 0.4 for MPE-tuned Faster R-CNN and SSD, respectively.

Fish counting

Despite tuning, comparison of the true and predicted counts on the test data fold showed biases (Figure 7). Examination of the data suggested biases were associated with the size of the bounding boxes. For both small and large bounding boxes the variance in the predicted count increased as the mean increased; however, the actual relationship between predicted and true count differed depending on the box size. This suggested that fish size or overlap between fish influenced estimation.

Figure 8 shows the results of statistical corrections calculated using the test data. For each of the optimal models selected during model confidence threshold tuning (either F1 or MPE tuning), D^2 increased and MAPE decreased when a Poisson correction model was added with the raw predicted count as features, and further increased when a correction for box size was also added. For the test data the evaluation of MPE-tuned SSD

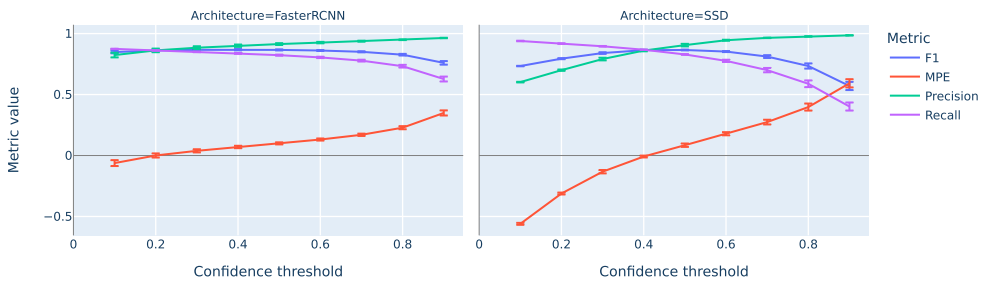


Figure 6. Evaluation of confidence thresholds using F1, MPE (mean percentage error), precision, and recall metrics on the training folds.

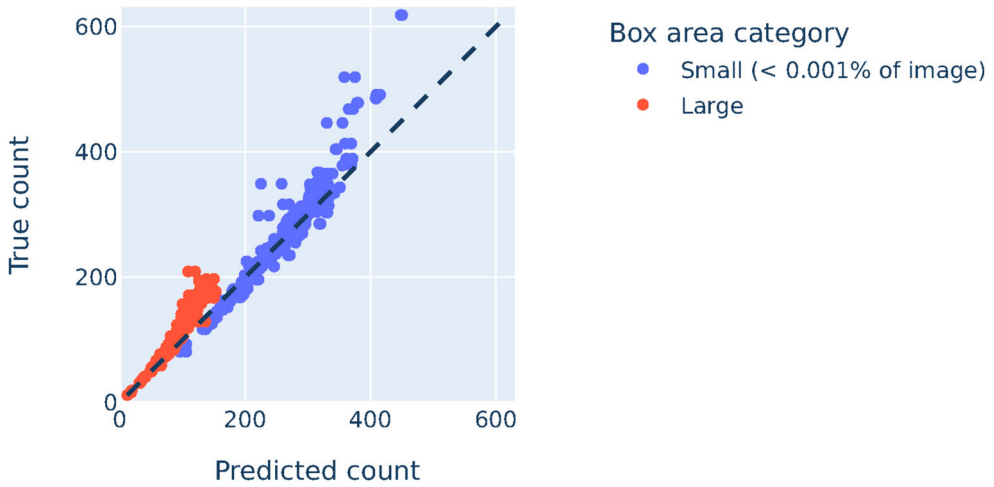


Figure 7. Actual vs. predicted counts for the F1-tuned Faster R-CNN model showing bimodal behaviour associated with box size, a proxy for fish age on the training folds. Actual counts were derived from counts of the bounding boxes applied during manual label annotation of images.

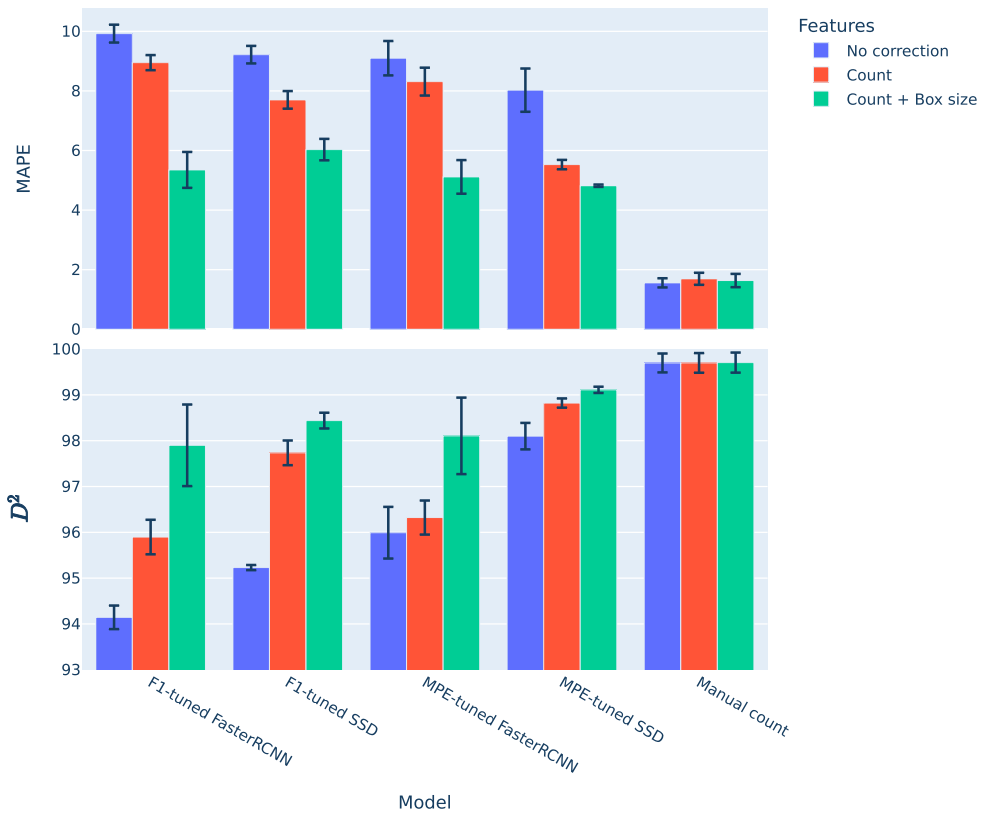


Figure 8. Mean absolute percentage error (MAPE) and deviance squared (D^2) for each thresholded object detection model combined with each set of features for the Poisson correction model, on the test folds. The Poisson correction model either had the raw predicted count or the raw predicted count plus box area as features.

Table 2. Count Estimate evaluation comparison between SSD, Faster R-CNN and Manual counts calculated using the test folds.

Model	Features	Mean MAPE	STD MAPE	Mean D^2	STD D^2
F1-tuned FasterRCNN	No correction	9.923299	0.299792	94.14404	0.257038
F1-tuned FasterRCNN	Count	8.949331	0.253394	95.89687	0.376965
F1-tuned FasterRCNN	Count + Box size	5.349279	0.602887	97.89961	0.890595
F1-tuned SSD	No correction	9.216894	0.294025	95.2305	0.056165
F1-tuned SSD	Count	7.699626	0.295538	97.7353	0.269661
F1-tuned SSD	Count + Box size	6.030539	0.359601	98.43926	0.17193
MPE-tuned FasterRCNN	No correction	9.098381	0.576865	95.99155	0.563955
MPE-tuned FasterRCNN	Count	8.312315	0.467646	96.32243	0.37179
MPE-tuned FasterRCNN	Count + Box size	5.114355	0.564129	98.10591	0.835107
MPE-tuned SSD	No correction	8.025924	0.725994	98.09984	0.288988
MPE-tuned SSD	Count	5.527671	0.157331	98.82238	0.100708
MPE-tuned SSD	Count + Box size	4.817984	0.039546	99.11085	0.068114
Manual count	No correction	1.558165	0.154087	99.6977	0.206328
Manual count	Count	1.694189	0.202058	99.69888	0.214771
Manual count	Count + Box size	1.635933	0.223024	99.70537	0.219284

Bolded values show the best mean D^2 or lowest MAPE for manual and counts estimated by manual count and by the best model.

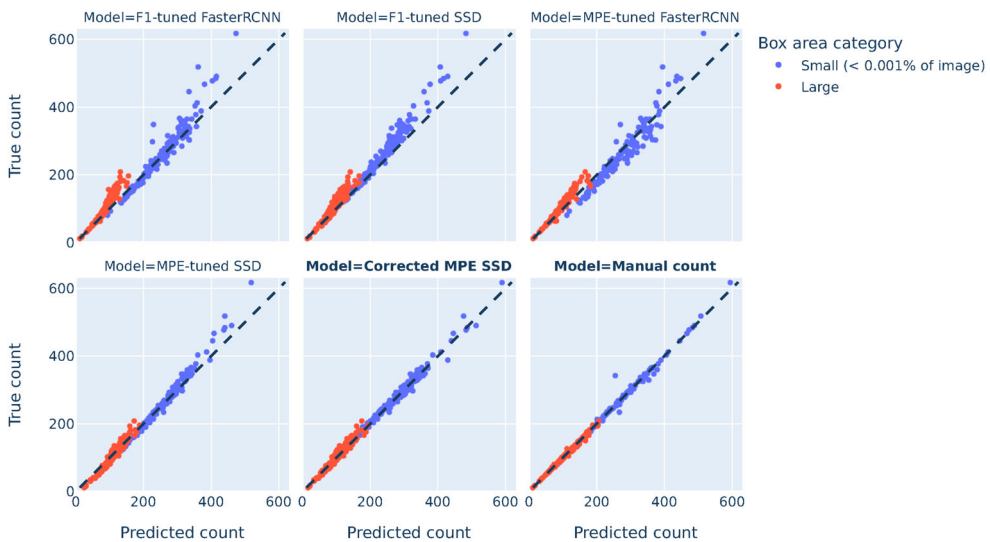


Figure 9. Comparison of true vs. predicted counts between models on the testing folds. Corrected MPE SSD refers to the MPE-tuned SSD model corrected with a Poisson regression using the raw predicted count and box area category features.

including a Poisson regression with both the raw predicted count and the box area as features was almost as accurate as that obtained from traditional manual counts (D^2 of 99.11% vs. D^2 of 99.71%, and MAPE of 4.82 vs. 1.64; Table 2 and Figures 8 and 9).

Discussion

Automatic object counting is receiving increasing attention in diverse fields, including cell and egg counting, traffic vehicle flow monitoring, pedestrian density warning, and

wildlife abundance and diversity estimation in reserves. However, counting in aquaculture is usually achieved manually, or using statistical methods, which is both time-consuming and tedious, and hence recent work has been exploring the possibility to apply automated object counting to assist fish breeding and production programmes (Saberioon et al. 2017; Li et al. 2021; Yang et al. 2021). The present work explored two state-of-the-art machine learning models for counting juvenile snapper in images, and below we discuss the success at implementing the models, including considerations about the speed of computation, then we compare the results from the models to those generated using manual methods. Finally, we conclude by summing our results up and by making suggestions about future applications and associated challenges.

Both SSD and Faster R-CNN models performed well at locating objects within the images, particularly given the challenges presented by the dataset (e.g. high overlap of objects). We were able to achieve deviance values for MPE-tuned SSD models similar to those obtained by manual counts ($D^2 > 95\%$); however, the MAPE for these models was still much higher than manual counts (MAPE $< 4.8\%$ for MPE-tuned SSD cf. $< 1.64\%$ for manual counts; Table 2). This indicates that there is more under- and over-counting than occurs with manual counts. Similarly, the F1 tuned models performed at greater than 90% D^2 on the test fold data, with MAPE less than 10% (Table 2).

We would like to clarify that automatic counts do not need to achieve 100% accuracy when compared to actual counts. Instead, a typically lower accuracy is acceptable and where this accuracy threshold lies is an application-dependent question, and needs to be decided on with the operators to ensure the application is fit for purpose. In our case, 5% errors or lower was deemed to be in the acceptable range as gains due to time savings would outweigh the minimal gains in errors. However, in other scenarios, where accuracy has a more pertinent nature (e.g. counting deformities), then more stringent metrics may be applied towards this goal. The main obstacle to increase the accuracy is related to the small size of the objects being detected and the high degree of overlap between objects. Both trained SSD and Faster-RCNN models detected small fish reasonably well, and the final MAPE was low ($< 10\%$) and with a $D^2 > 94\%$. For both models, when fish overlapped greatly in the image due to high fish density, then the prediction error rate increased, as predicted by previous research (Connolly et al. 2021).

The higher error rate of automated as opposed to manual counts for many of our final models does represent a challenge for adoption in an applied setting, and therefore further work may be required to improve accuracy. While errors of less than 5% may seem sufficient for a data scientist, this likely means a different thing to workers at facilities who are making decisions regarding holding densities and feed amounts, and where differences of 5% can thus significantly impact on animal welfare and operational costs. Methodological adjustments during fish counting that help to minimise overlap of individuals would increase the prediction accuracy and so real-life testing with end users remains a crucial next step to ensure further refinements and adoption of this method.

Overlap of objects, or occlusion, is a general problem encountered by automated methods and considerable tuning and statistical corrections were needed here to help overcome errors that result from this occlusion. A recent paper on the same species as investigated here examined video footage derived from baited remote underwater video stations (BRUVS) to develop an automated and repeatable deep learning method for monitoring relative abundance of snapper to support stock assessment

models of this species (Connolly et al. 2021). Occlusion due to fish overlap created a similar problem in their study. Similar to our study, the authors tested combinations of varying confidence thresholds, on/off use of sequential non-maximum suppression (Seq-NMS), and inclusion of a statistical correction step to further optimise their modelling approach. They found that a combination of Seq-NMS, a 55% confidence threshold, and a cubic polynomial corrective equation provided the best predictive accuracy at high densities. In our study, the inclusion of both a Poisson regression of the predicted count and the box size provided a similar result. However, other corrective equations may be needed for this study to further refine the modelling approach in the future and to increase the accuracy of automated counts, while decreasing costs.

One of the major advantages of machine learning methods is the ability to automate and increase the efficiency of methods. The testing and tuning of these computer vision pipelines took a few months to establish and build into a stable architecture. However, after this initial investment in time, a minimal amount of time is required to maintain the pipeline and to perform checks and upgrades to improve and refine model performance. Thus, taken together, when comparing our machine learning model performance to the time and effort required to undertake the manual counting, we found that while the manual setup requires comparatively minimal investment and infrastructure (e.g. bins and staff trained in carrying out manual counts), any subsequent counting work necessitates an investment of roughly 1–3 h for each batch counted (dependent on fish numbers). We would also like to note that there may be a trade-off between manual labour time and accuracy levels achieved, i.e. with increasing manual labour time it is foreseeable that the accuracy of counting would decline in parallel, as has been observed when conducting other fish counting methods (Sale and Douglas 1981).

To reach harvest size in species grown in aquaculture, species like snapper are typically grown for 2–3 years, and to achieve this, staff are required to move fish multiple times as they grow (Garcia Garcia et al. 2011). Knowing the fish density in a tank is also necessary when estimating the feed amount required, reducing inter-individual aggression or when estimating the water flow needed to ensure high oxygen availability and the removal of waste products (Muir 2005; Ashley 2007). Given the frequent need of counting fish, particularly when individuals are young and thus in need to be moved more frequently, the time savings achieved by implementing the automated model would sum up quite considerably. In addition to the need of knowing fish biomass, data on individual morphological performance in terms of growth achievements and deformities are also of major importance in aquaculture. Images pave the way towards this goal, as once they have been taken they facilitate additional measurements to be taken at minimal costs in terms of time. Thus, future coupling of fish counting software and morphometric tools holds significant value for aquaculture breeding programmes, and also production facilities, as they can provide rapid insights into growth metrics and deformity ratios of fish cohorts. Further improvements and associated gains can be made if image inferencing can be run at point of capture using simple hardware, such as an off-the-shelf GoPro camera or a mobile phone. Data can then be returned live and this can reduce the number of grading and sorting operations needed as real time decision making is applied during these operations. We estimate total staff hours needed for counting and grading operations in our hatchery would reduce to at least half if live count data were available.

Our study adds to the mounting evidence showing that computer vision technologies, as a non-invasive method, could be used to count the number of objects in aquaculture effectively and quickly (Saberioon et al. 2017). The use of machine learning for object detection and counting in images has been growing rapidly in recent years, with frequent advances and new approaches (Ge et al. 2020). However, when transferring these methods to live animals in an aquaculture setting, specific issues arise that need careful consideration (Bochkovskiy et al. 2020 Apr 23; Nguyen et al. 2020; Solawetz 2020). Specifically, challenges caused by the objects themselves, such as the transparency of the object, the difference in shape and size of the object, and the overlapping object issues caused by the high fish densities and/or fish movement. Moreover, additional difficulties can arise from the complexity of the background environment, such as interference issues, the disturbance of water flow, and the complexity of the tank or seabed environment. Future focus needs to be on improving the accuracy of models to get closer to or matching those achieved by manual counts, as well as coupling the methods to new hardware developments to both enable real time data capture as well as to improve data capture quality (e.g. new types of bins or chutes) (Li et al. 2020). Enabling real-time deployment and decision making would be particularly valuable, contributing to reduced operational costs of aquaculture operations.

A further improvement to the method would be setting up the imaging equipment to take multiple images, potentially using burst capture or as a short video clip. Each frame could be processed separately, with averaging applied, or video tracking could be implemented to resolve ambiguities between frames. This would improve the accuracy at the expense of moderate increases in processing power and time.

Another small variation of the documented technique would be to reduce the overall fish density in bins for future use, because while this would increase the number of bins that would have to be carried by staff, it could also considerably increase the accuracy of the model by reducing the overlap of fish, and thus objects that need to be counted. Furthermore, in addition to counting fish, additional measurements could be retrieved from the same image data (e.g. size, disease, deformities) to yield added insights into fish welfare and performance. This would be particularly significant for selective breeding programmes, where selection of superior individuals is used to generate elite lines that show improved performance traits (Murata et al. 1996).

Conclusion

The models developed in this study were able to identify most juvenile fish objects in the bin images taken in the hatchery. While the models did not get down to the error rate of that of manual counts, we were able to reduce errors to less than 10% and less than 5% in MPE – tuned models. In addition, our documented method added a large increase in processing speed, thus facilitating throughput and decreasing economic costs. Based on the training dataset we identified the best models using R-CNN or SSD architecture tuned either by MPE or F1. When evaluated on the test dataset the MPE-tuned SSD models in which a bias correction including a Poisson regression of initial predicted counts and a parameter estimating box size, we suggest that this model is likely to be most accurate for processing fish within the Plant and Food fish facility. However, in other facilities or if fish sizes fell outside the distribution of the current dataset, we

suggest that further evaluation and monitoring of the counts is needed. It is very likely that these tuning parameters may be specific to particular characteristics of the current data.

We have highlighted a number of future directions which could be pursued with this work to improve and enhance its use and to allow greater levels of information to be collected about aquaculture populations and their performance. Although this application was developed specifically for use with fish larvae, it should be useful for older life stages of fish (and possibly other animals) for which satisfactory images are available. In addition, the machine learning work carried out in this project is built around a generalisable architecture, which has an additional benefit of enabling projects across a wide diversity of problems from fish to plants.

Acknowledgements

We would like to acknowledge the help of multiple staff at PFR that have contributed to this project and other pieces of research around the extraction and analyses of imaging data which lead to this work being possible. This project was funded by the New Zealand Government via the Ngā Pou Rangahau platform, a research framework developed by the New Zealand Institute for Plant and Food Research Limited and supported by a Strategic Science Investment Fund grant RTVU1914 from the Ministry of Business, Innovation and Employment.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This project was funded by the New Zealand Government through the Ministry of Business, Innovation, and Employment (MBIE) SSIF grant C11X1702 and supported by an additional MBIE SSIF grant for Data Science under Contract VUW RTVU1914.

ORCID

Krishna Moorthy Babu  <http://orcid.org/0000-0002-4096-5320>

Daniel Bentall  <http://orcid.org/0000-0002-3537-7073>

David T. Ashton  <http://orcid.org/0000-0002-0996-6368>

Morgan Puklowski  <http://orcid.org/0000-0003-2961-2177>

Warren Fantham  <http://orcid.org/0000-0001-7688-7670>

Harris T. Lin  <http://orcid.org/0000-0002-3138-1219>

Nicholas P. L. Tuckey  <http://orcid.org/0000-0003-1437-636X>

Maren Wellenreuther  <http://orcid.org/0000-0002-2764-8291>

Linley K. Jesson  <http://orcid.org/0000-0003-4969-160X>

References

- Antonucci F, Costa C. 2020. Precision aquaculture: a short review on engineering innovations. *Aquaculture International*. 28(1):41–57. doi:10.1007/s10499-019-00443-w.
- Ashley PJ. 2007. Fish welfare: current issues in aquaculture. *Applied Animal Behaviour Science*. 104(3–4):199–235.

- Ashton D, Ritchie P, Wellenreuther M. 2019. High-density linkage map and QTLs for growth in snapper (*Chrysophrys auratus*). *G3 Genes|Genomes|Genetics*. 9(4):1027–1035. doi:10.1534/g3.118.200905.
- Baesjou J-P, Wellenreuther M. 2021. Genomic signatures of domestication selection in the Australasian snapper (*Chrysophrys auratus*). *Genes*. 12(11):1737. doi:10.3390/genes12111737.
- Béné C, Barange M, Subasinghe R, Pinstrup-Andersen P, Merino G, Hemre G-I, Williams M. 2015. Feeding 9 billion by 2050—putting fish back on the menu. *Food Security*. 7(2):261–274.
- Bochkovskiy A, Wang C-Y, Liao H-YM. 2020. YOLOv4: optimal speed and accuracy of object detection. [accessed 2022 May 12]. <https://arxiv.org/abs/2004.10934>. doi:10.48550/ARXIV.2004.10934.
- Connolly RM, Fairclough DV, Jinks EL, Ditria EM, Jackson G, Lopez-Marcano S, Olds AD, Jinks KI. 2021. Improved accuracy for automated counting of a fish in baited underwater videos for stock assessment. *Front Mar Sci*. 8:658135. doi:10.3389/fmars.2021.658135.
- Everingham M, Van Gool L, Williams CKI, Winn J, Zisserman A. 2010. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*. 88(2):303–338.
- FAO. 2018. Meeting the sustainable development goals. Rome (The state of world fisheries and aquaculture).
- Fu G, Yuna Y. 2022. Phenotyping and phenomics in aquaculture breeding. *Aquaculture and Fisheries*. 7(2):140–146. doi:10.1016/j.aaf.2021.07.001.
- Gao G, Gao J, Liu Q, Wang Q, Wang Y. 2020. CNN-based density estimation and crowd counting: a survey. [accessed 2022 May 12]. <https://arxiv.org/abs/2003.12783>. doi:10.48550/ARXIV.2003.12783.
- Garcia Garcia B, Basurco B, Lovatelli A. 2011. Current status of Sparidae aquaculture. In *Sparidae: Biology and aquaculture of gilthead sea bream and other species*. p. 1–50.
- Ge C, Jing W, Jingyu W, Qi Q, Sun H, Liao J. 2020. Towards automatic visual inspection: A weakly supervised learning method for industrial applicable object detection. *Computers in Industry*. 121:103232. doi:10.1016/j.compind.2020.103232.
- Gjedrem T, Robinson N. 2014. Advances by selective breeding for aquatic species: a review. *Agricultural Sciences*. 5(12):1152–1158. doi:10.4236/as.2014.512125.
- Godfray HCJ, Beddington JR, Crute IR, Haddad L, Lawrence D, Muir JF, Pretty J, Robinson S, Thomas SM, Toulmin C. 2010. Food security: the challenge of feeding 9 billion people. *Science*. 327(5967):812–818.
- Gupta S, Kumar M, Garg A. 2019. Improved object recognition results using SIFT and ORB feature detector. *Multimedia Tools and Applications*. 78(23):34157–34171.
- Li D, Hao Y, Duan Y. 2020. Nonintrusive methods for biomass estimation in aquaculture with emphasis on fish: a review. *Reviews in Aquaculture*. 12(3):1390–1411.
- Li D, Miao Z, Peng F, Wang L, Hao Y, Wang Z, Chen T, Li H, Zheng Y. 2021. Automatic counting methods in aquaculture: a review. *Journal of the World Aquaculture Society*. 52(2):269–283. doi:<https://doi.org/10.1111/jwas.12745>.
- Lin T-Y, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL. 2014. Microsoft COCO: common objects in context. In: Fleet D, Pajdla T, Schiele B, Tuytelaars T, editors. *Computer vision – ECCV 2014*. Vol. 8693. Cham: Springer International Publishing; pp. 740–755. [accessed 2022 May 12]. http://link.springer.com/10.1007/978-3-319-10602-1_48.
- Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu C-Y, Berg AC. 2016. SSD: single shot MultiBox detector. In: Leibe B, Matas J, Sebe N, Welling M, editors. *Computer vision – ECCV 2016*. Vol. 9905. Cham: Springer International Publishing; pp. 21–37. [accessed 2022 May 12]. http://link.springer.com/10.1007/978-3-319-46448-0_2.
- Mahanty M, Bhattacharyya D, Midhunchakkaravarthy D. 2022. A review on deep learning-based object recognition algorithms. In: Bhattacharyya D, Saha SK, Fournier-Viger P, editors. *Machine intelligence and soft computing*. Vol. 1419. Singapore: Springer Nature Singapore. (Advances in intelligent systems and computing); pp. 53–59. [accessed 2022 May 12]. https://link.springer.com/10.1007/978-981-16-8364-0_7.

- Mathiassen JR, Misimi E, Bondø M, Veliyulin E, Østvik SO. 2011. Trends in application of imaging technologies to inspection of fish and fish products. *Trends in Food Science & Technology*. 22 (6):257–275.
- Muir J. 2005. Managing to harvest? Perspectives on the potential of aquaculture. *Philosophical Transactions of the Royal Society B: Biological Sciences*. 360(1453):191–218. doi:10.1098/rstb.2004.1572.
- Murata O, Harada T, Miyashita S, Izumi K, Maeda S, Kato K, Kumai H. 1996. Selective breeding for growth in red sea bream. *Fisheries Science*. 62(6):845–849.
- Nguyen N-D, Do T, Ngo TD, Le D-D. 2020. An evaluation of deep learning methods for small object detection. *Journal of Electrical and Computer Engineering*. 2020:1–18.
- Rasmussen JH, Moyano M, Fuiman LA, Oomen RA. 2022. Fishsizer: software solution for efficiently measuring larval fish size. *Ecology and Evolution*. 12(3). [accessed 2022 May 12]. <https://onlinelibrary.wiley.com/doi/10.1002/ece3.8672>. doi:10.1002/ece3.8672.
- Ren S, He K, Girshick R, Sun J. 2015. Faster R-CNN: towards real-time object detection with region proposal networks. In: *Proceedings of the 28th International Conference on Neural Information Processing Systems – Volume 1*. Cambridge, MA, USA: MIT Press. (NIPS'15). p. 91–99.
- Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, et al. 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*. 115(3):211–252. doi:10.1007/s11263-015-0816-y.
- Saberioon M, Gholizadeh A, Cisar P, Pautsina A, Urban J. 2017. Application of machine vision systems in aquaculture with emphasis on fish: state-of-the-art and key issues. *Reviews in Aquaculture*. 9(4):369–387.
- Sale PF, Douglas WA. 1981. Precision and accuracy of visual census technique for fish assemblages on coral patch reefs. *Environmental Biology of Fishes*. 6(3):333–339. doi:10.1007/BF00005761.
- Sekachev B, Manovich N, Zhiltsov M, Zhavoronkov A, Kalinin D, Hoff B, TOSmanov, Kruchinin D, Zankevich A, DmitriySidnev, et al. 2020. opencv/cvat: v1.1.0. Zenodo. [accessed 2022 May 12]. <https://zenodo.org/record/4009388>.
- Solawetz J. 2020. Tackling the small object problem in object detection. <https://blog.roboflow.com/detect-small-objects/>.
- Valenza-Troubat N, Hilario E, Montanari S, Morrison-Whittle P, Ashton D, Ritchie P, Wellenreuther M. 2022. Evaluating new species for aquaculture: A genomic dissection of growth in the New Zealand silver trevally (*Pseudocaranx georgianus*). *Evolutionary Applications*. 15(4):591–602. doi:10.1111/eva.13281.
- Wang R, Shi Y, Cao W. 2019. GA-SURF: A new speeded-up robust feature extraction algorithm for multispectral images based on geometric algebra. *Pattern Recognition Letters*. 127:11–17.
- Yang L, Liu Y, Yu H, Fang X, Song L, Li D, Chen Y. 2021. Computer vision models in intelligent aquaculture with emphasis on fish detection and behavior analysis: A review. *Archives of Computational Methods in Engineering*. 28(4):2785–2816.
- Zhang Y, Zhang F, Cheng J, Zhao H. 2021. Classification and recognition of fish farming by extraction new features to control the economic aquatic product. *Complexity*. 2021:1–9. doi:10.1155/2021/5530453.