

ICES Journal of Marine Science

Journal du Conseil

Volume 77 Number 4 July/August 2020

<http://academic.oup.com/icesjms>

Contents

Applications of machine learning and artificial intelligence in marine science

1267 Introduction

C. Beyan & H. I. Browman

Setting the stage for the machine intelligence era in marine science

1274 Quo Vadimus

K. Malde, N. O. Handegard, L. Eikvil, & A.-B. Salberg

Machine intelligence and the data-driven future of marine science

1286 *W. N. Probst*

How emerging data technologies can increase trust and transparency in fisheries

1295 Original Articles

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

1308 *A. Mahmood, M. Bennamoun, S. An, F. Sohel, F. Boussaid, R. Hovey, & G. Kendrick*

Automatic detection of Western rock lobster using synthetic data

1318 *Y.-C. Lu, C. Tung, & Y.-F. Kuo*

Identifying the species of harvested tuna and billfish using deep convolutional neural networks

1330 *A. Álvarez-Ellacuría, M. Palmer, I. A. Catalán, & J.-L. Lisani*

Image-based, unsupervised estimation of fish size from commercial landings using deep learning

1340 *G. French, M. Mackiewicz, M. Fisher, H. Holah, R. Kilburn, N. Campbell, & C. Needle*

Deep neural networks for analysis of fisheries surveillance video and automated monitoring of fish discards

1354 *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

1367 *C.-H. Tseng & Y.-F. Kuo*

Detecting and counting harvested fish and identifying fish types in electronic monitoring system videos using deep convolutional neural networks



- 1379** R. Proud, R. Mangeni-Sande, R. J. Kayanda, M. J. Cox, C. Nyamweya, C. Ongore, V. Natugonza, I. Everson, M. Alison, L. Hobbs, B. B. Kashindy, E. W. Mlaponi, A. Taabu-Munyaho, V. M. Mwainge, E. Kagoya, A. Pegado, E. Nduwayesu, & A. S. Brierley
Automated classification of schools of the silver cyprinid *Rastrineobola argentea* in Lake Victoria acoustic survey data using random forests
- 1391** O. Brautaset, A. U. Waldeland, E. Johnsen, K. Malde, L. Eikvil, A.-B. Salberg, & N. O. Handegard
Acoustic classification in multifrequency echosounder data using deep convolutional neural networks
- 1401** N. Semmar & A. M. Vaz-dos-Santos
Highlighting growth regulation processes in fish populations by a simplex simulation approach: application to *Merluccius hubbsi* stocks in the Southwestern Atlantic
- 1414** C. Liu, S. Zhou, Y.-G. Wang, & Z. Hu
Natural mortality estimation using tree-based ensemble learning models
- 1427** Q. Li, X. Sun, J. Dong, S. Song, T. Zhang, D. Liu, H. Zhang, & S. Han
Developing a microscopic image dataset in support of intelligent phytoplankton detection using deep learning
- 1440** R. W. Campbell, P. L. Roberts, & J. Jaffe
The Prince William Sound Plankton Camera: a profiling *in situ* observatory of plankton and particulates



Introduction to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Introduction

Setting the stage for the machine intelligence era in marine science

Cigdem Beyan ^{1*} and Howard I. Browman ²

¹Pattern Analysis and Computer Vision, Istituto Italiano di Tecnologia, Via Enrico Melen 83, Genova 16152, Italy

²Institute of Marine Research, Ecosystem Acoustics Group, Austevoll Research Station, Sauganeset 16, Storebø N-5392, Norway

*Corresponding author: tel: +39 346 7664 779; e-mail: cigdem.beyan@iit.it.

Beyan, C. and Browman, H. I. Setting the stage for the machine intelligence era in marine science. – ICES Journal of Marine Science, 77: 1267–1273.

Received 18 April 2020; revised 18 April 2020; accepted 20 April 2020; advance access publication 18 June 2020.

Machine learning, a subfield of artificial intelligence, offers various methods that can be applied in marine science. It supports data-driven learning, which can result in automated decision making of *de novo* data. It has significant advantages compared with manual analyses that are labour intensive and require considerable time. Machine learning approaches have great potential to improve the quality and extent of marine research by identifying latent patterns and hidden trends, particularly in large datasets that are intractable using other approaches. New sensor technology supports collection of large amounts of data from the marine environment. The rapidly developing machine learning subfield known as deep learning—which applies algorithms (artificial neural networks) inspired by the structure and function of the brain—is able to solve very complex problems by processing big datasets in a short time, sometimes achieving better performance than human experts. Given the opportunities that machine learning can provide, its integration into marine science and marine resource management is inevitable. The purpose of this themed set of articles is to provide as wide a selection as possible of case studies that demonstrate the applications, utility, and promise of machine learning in marine science. We also provide a forward-look by envisioning a marine science of the future into which machine learning has been fully incorporated.

Keywords: analysis of underwater acoustics data, artificial intelligence, computer vision, data processing, deep learning, machine learning, surveillance and inspection of fish catch, underwater image analysis

Background and motivation for this themed article set

Artificial intelligence (AI) is increasingly being applied to all kinds of data. Some applications of AI are face recognition systems, natural language processing (e.g. speech recognition, language understanding, language generation, and language translation), disease detection systems, video surveillance, quality inspection in manufacturing, product design and creation, robotics, and self-driving cars (Dargan *et al.*, 2019). It is accurate to say that AI is now everywhere, from our smartphones, to web-browser, to cars.

Machine learning (ML), which is a subfield of AI, implements dynamic models resulting in data-driven decisions. ML techniques can be applied to high-dimensional (Fan *et al.*, 2009), nonlinear, complex, and big data. Further, the ML approach is effective even in cases where the data are noisy (e.g. Frenay and Verleysen, 2014; Xiao *et al.*, 2015) or some identification labels are missing (McKnight *et al.*, 2007; Aste *et al.*, 2015). ML is also able to address the small sample size problem: so-called zero or few-shot learning (Huo *et al.*, 2019). What makes ML most appealing is its capacity to handle problems that are impossible or too challenging for traditional approaches, which require many

people and considerable time and resources to produce the desired accuracy. In other words, ML provides not only effective solutions, robustness, and accuracy but also efficiency as it can rapidly process huge amounts of data.

Deep learning (DL), inspired by the structure and function of the human brain, is a subfield of ML that involves the use of artificial neural networks (ANNs). ANN can take several forms, including recurrent neural networks (Hochreiter and Schmidhuber, 1997) and convolutional neural networks (CNNs) (Krizhevsky *et al.*, 2012). Although ANNs are not new, their wide use only became practical after the development of massively parallel graphical processing units (GPUs). GPUs provide computation power and fast processing so that DL architectures running on GPUs can analyse huge amounts of data quickly and efficiently. In 2012, Krizhevsky *et al.* (2012) proved that CNNs can achieve a high level of accuracy in image classification. The success of CNNs has been extended to other computer vision tasks, for example object localization (Ren *et al.*, 2015; Redmon *et al.*, 2016), semantic segmentation (Long *et al.*, 2015; Badrinarayanan *et al.*, 2017), natural language processing for speech recognition (Hinton *et al.*, 2012), machine translation (Sutskever *et al.*, 2014), optical character recognition (Goodfellow *et al.*, 2014), face recognition and verification (Taigman *et al.*, 2014), object recognition (Xiao *et al.*, 2015), and so forth. All of these will, in due course, be applied to data analysis in many branches of research, including marine science.

Studying and sustainably managing marine ecosystems presents special challenges because they are three dimensional, expansive, very dynamic, and complex. These characteristics require data collection over a wide range of spatiotemporal scales, which has been a major challenge (Godø *et al.*, 2014; Janzen *et al.*, 2019). Rapid progress in sensors, information, and communication technologies now allows marine scientists to collect large volumes of data at ever lower cost.

“Cruises now regularly return to port with terabytes of data, high temporal resolution coastal time series contain billions of measurements, and water samples are parsed into millions of DNA sequences.” (POGO Workshop, 2019)

Moored buoys support long-term monitoring and high resolution measurements of physical, chemical, and biological variables, as well as acoustics, at fixed locations and transmit their data in real-time via satellite uplink or cabled connection to shore (e.g. Aguzzi *et al.*, 2015; Van Engeland *et al.*, 2019). However, they are limited to monitoring depths from the seabed to the ocean surface. Sophisticated and heavily instrumented towed observation platforms, and autonomous drones, are collecting large volumes of data of many types (e.g. De Robertis *et al.*, 2019; Lombard *et al.*, 2019; Verfuss *et al.*, 2019). However, the capacity of human experts to filter, curate, and analyse all of these data is limited. This is where ML and AI will be making greater-and-greater contributions as methods improve and are implemented more broadly.

ML can be applied to automate various routine tasks in marine science. The prediction of ocean weather, for example detecting sea surface temperature (Tanaka *et al.*, 2004; Wu *et al.*, 2006), habitat modelling (Krasnopolsky, 2009; Thessen, 2016), modelling monsoons (Cavazos *et al.*, 2002), forecasting sea level fluctuations (Makarynsky *et al.*, 2004), wind and wave modelling (Forget *et al.*, 2015; James *et al.*, 2018), and the detection of acute

situations, for example oil spill and other point sources of pollution (Kubat *et al.*, 1998) are just some of the applications. Continuous underwater video and acoustic surveillance systems are rapidly developing tools to monitor marine life while computer vision and ML techniques contribute by automatically analysing the massive data streams from these platforms (e.g. Fisher *et al.*, 2016). These data can already be used to extract higher-level interpretations by automatically detecting and tracking fish underwater (Spampinato *et al.*, 2008), identifying fish species (Joly *et al.*, 2015; Siddiqui *et al.*, 2018; Villon *et al.*, 2018; Allken *et al.*, 2019), and estimating swimming trajectories and speeds (Beyan *et al.*, 2018). Eventually, it will be possible to use time series of these data streams to assess changes in species abundance and distribution, environmental change, predator–prey relationships, and more (Fisher *et al.*, 2016). Baited cameras and camera traps allow data to be collected without disturbing animals, which produces high volumes of images that can be analysed by using DL techniques (e.g. Tabak *et al.*, 2019). There is also great potential to apply DL to automatic fish identification, counting, and sizing on fishing vessels (e.g. Bartholomew *et al.*, 2018).

In this context, the objective of this themed set of articles was to bring together contributions on the broad theme of the applications of AI, ML, DL, and advanced data systems (e.g. block chains) to research, monitoring and management of marine organisms and ecosystems. We sought contributions on the following topics, among others,

- Automatic marine ecosystem monitoring based on visual and/or acoustic data;
- Automatic fish detection;
- Automatic coral reef state detection (e.g. health, dead/alive);
- Underwater measurement of fish length;
- Automatic fish counting, for example to analyse the effect of global warming;
- Automatic fish tracking (e.g. swimming speeds and trajectories);
- Automatic fish species classification/recognition/identification;
- Characterizing interactions between fish (e.g. predator–prey relationships);
- Fine-grained automatic object recognition in underwater visual data (e.g. substrate classification, plankton);
- Applications of block chain technology/systems;
- Automatic detection/classification of acoustics produced by marine animals (e.g. whales, dolphins, and fish); and
- Automatic systems for fisheries management.

We received 30 submissions in response to the call for papers. The 15 that made it through the peer review process are described below. The articles that appear in this themed set, and the many relevant articles that they cite, demonstrate that AI is already a very helpful tool in a wide variety of applications in marine science.

The articles in this theme set

With the exception of Semmar and Vaz-dos-Santos (2019), Liu *et al.* (2020), and Proud *et al.* (2020), all of the articles present methods based on DL (mainly CNNs), or at least mention the advances in DL and its great potential. The articles can be categorized in terms of (i) the environment that has been examined,

that is unconstrained underwater (Mahmood *et al.*, 2019; Salman *et al.*, 2019), observing fish catch on fishing trawlers (French *et al.*, 2019; Garcia *et al.*, 2019; Tseng and Kuo, 2020), fishing vessels (e.g. Lu *et al.*, 2019), and fish caught in a box (Álvarez-Ellacuría *et al.*, 2019); (ii) the type of marine organisms investigated, that is fish (Álvarez-Ellacuría *et al.*, 2019; French *et al.*, 2019; Lu *et al.*, 2019; Malde *et al.*, 2019; Probst, 2019; Salman *et al.*, 2019; Brautaset *et al.*, 2020), lobster (Mahmood *et al.*, 2019), and plankton (Li *et al.*, 2019; Campbell *et al.*, 2020); (iii) the type of data used, that is images (Álvarez-Ellacuría *et al.*, 2019; French *et al.*, 2019; Garcia *et al.*, 2019; Lu *et al.*, 2019; Mahmood *et al.*, 2019; Malde *et al.*, 2019; Salman *et al.*, 2019; Campbell *et al.*, 2020; Lu *et al.*, 2020; Tseng and Kuo, 2020) and video or audio (Brautaset *et al.*, 2020; Proud *et al.*, 2020). These articles are summarized below.

Malde *et al.* (2019) review recent developments in ML, mainly DL, and stress the opportunities and challenges associated with integration of DL into marine science. Probst (2019) focuses on how blockchains, data mining and AI can improve trust between producers, wholesalers, retailers, consumers, management authorities, and scientists by increasing transparency and availability of information throughout the supply chain. It is claimed that these digital technologies can make the flow of money associated with the global stream of seafood products more visible and transparent.

Salman *et al.* (2019) propose a method that relies on region-based deep CNNs to detect freely moving fish in unconstrained underwater environments. Motion images obtained by applying Gaussian Mixture Models (Stauffer and Grimson, 1999; Zivkovic and Heijden, 2006) and the optical flow method (Beauchemin and Barron, 1995) are combined with raw greyscale video images. The resulting three-channel image is input to a CNN model, which detects fish. The proposed method was tested on two datasets composed of 42 493 and 1328 labelled fish and produced a state-of-the-art performance for underwater fish detection. The experimental analysis was performed on videos that included several real-world challenges such as blurred images, complex and dynamic backgrounds, crowded scenes, and luminosity changes. Mahmood *et al.* (2019) also focus on automatic underwater image analysis, although the focal species was western rock lobster (*Panulirus cygnus*). The method proposed by Mahmood *et al.* (2019) is also based on DL. The authors note that detection of the rock lobster faces the challenge of having little annotated data available. To handle this, a synthetic dataset was generated that was used to fine-tune the state-of-the-art object detector, YOLOv3 (Redmon and Farhadi, 2018), for detection of rock lobster. Unusually, the individual body parts rather than the whole animal were synthesized. YOLOv3 was trained using the synthetic data only and the resulting model was tested on real-world images. This training scheme showed significantly improved results compared with using real-world images in training and testing. Mahmood *et al.* (2019) highlight the fact that, for many marine animals, the amount of labelled data is still limited. Despite that, they demonstrate that DL technology can be effective even when the amount of data available is limited.

Lu *et al.* (2019) propose a method that models the images of fish on the decks of fishing vessels to identify them to species. The species included in their study were albacore (*Thunnus alalunga*), bigeye tuna (*Thunnus obesus*), yellowfin tuna (*Thunnus albacares*), blue marlin (*Makaira nigricans*), Indo-pacific sailfish (*Istiophorus platypterus*), and swordfish (*Xiphias gladius*). A pre-

trained VGG-16 model (Simonyan and Zisserman, 2015) was fine-tuned for fish species identification and showed a high overall accuracy. Besides the quantitative results, the image regions detected as informative for the classification task were also characterized. Performing such qualitative analysis is important because it provides an explanation and interpretation of the function of the trained CNN model. Another DL-based method applied to fish catches, with the aim of measuring fish length automatically, is presented by Álvarez-Ellacuría *et al.* (2019). Conventionally, fish lengths have been manually calculated for a small number of randomly selected fish. The method proposed by these authors first applies Mask R-CNN (He *et al.*, 2017) to the images of European hake (*Merluccius merluccius*) displayed in boxes (containers used at points of sale holding many fish inside) to segment the fish heads. Image segmentation is the process of partitioning an image into multiple segments (e.g. image objects) that are composed of sets of pixels. A statistical model is then used to estimate the total fish length from the length of the segmented fish heads.

French *et al.* (2019) developed a computer vision system designed to monitor and quantify the fish that are discarded on fishing trawlers. The system is accurate and robust even when the orientation of the fish is variable, when there are occlusions among fish and when there are occlusions in the working area (e.g. from fishers processing the catch). The instance segmentation (the task of detecting and delineating each distinct object of interest in an image) component is based on separate Mask R-CNN models (He *et al.*, 2017), a different one for each conveyor belt. The segmented fish are passed to a CNN-based species classifier. The system was evaluated in four different settings: (i) using only the research samples composed of a large number of training samples, uniform lighting, uniform appearance, and less occlusions. This setting provides an upper bound for the performance of the system; (ii) using only the commercial samples such that training and testing samples are considerably fewer but the conditions are more challenging than the previous scenario, that is resulting in worse performance; (iii) applying leave-one-belt-out cross validation such that training was applied on samples from some commercial belts and the research samples, whereas testing was performed using samples from a never seen commercial belt. This is the case that is closest to the real-world scenario; and (iv) training on research samples and testing on commercial samples. This is the most challenging scenario for a classifier because of the domain gap (the situation that arises when the data distribution across different domains are dissimilar). However, it is also the most ideal scenario to prepare training data because little effort is required for annotation. Additionally, a comparison between the species identification component and human experts was conducted. Human experts achieved a mean class accuracy of 74–86%, whereas the DL classifier achieved ~58%, which is slightly better than the poorest human expert.

Garcia *et al.* (2019) also present a method to perform automatic fish segmentation and fish size measurement, although they use stereo images acquired using an imaging system placed in the trawl. Assuming that stereo imaging can increase the robustness and accuracy of fish length measurements (French *et al.*, 2019), a Mask R-CNN model (He *et al.*, 2017) is used to localize and segment individual fish in an image. Unlike French *et al.* (2019), the proposed pipeline applies a preprocessing step, which tries to reduce domain gaps that might arise from, for example those resulting from variability in the background illumination and

differences in appearance of the fish in different datasets. Additionally, a post-processing step, which performs a gradient-based boundary estimation given the Mask R-CNN's results as the inputs, is applied to provide more accurate boundaries. The proposed fish localization pipeline performs well even in highly cluttered images containing overlapping fish.

Tseng and Kuo (2020) propose an approach for pre-screening harvested fish in videos from electronic monitoring systems (EMS). Using a Mask R-CNN model (He *et al.*, 2017), the harvested fish in the frames of the EMS videos are segmented from the background. The fish are counted using time thresholding (to remove false-positive detections) and distance thresholding (if the distance is less than a threshold the candidate fish identified are considered the same in order to avoid recounting the same fish in sequential frames). Subsequently, the types and body lengths of the fish are determined using the Mask R-CNN model's confidence score. The videos were acquired under uncontrolled weather conditions (e.g. sunny days, rainy days, and dark nights). A total of 500 videos were used for training and validation of the Mask R-CNN model (He *et al.*, 2017) for fish detection and segmentation. The remaining 200 videos were used for assessing the proposed fish counting method. The trained Mask R-CNN model resulted in a recall of 97.58% and a mean average precision of 93.51% for fish detection. For fish counting, a recall of 93.84% and a precision of 77.31% were obtained. Additionally, for fish type identification, an overall accuracy of 98.06% was obtained.

Proud *et al.* (2020) apply an automated method to identify echoes from Daga schools (*Rastrineobola argentea*) in echo sounder data collected during fish stock-assessment surveys in Lake Victoria. Only the acoustic data collected between sunrise and sunset were analysed. A random forest (RF) classifier was constructed using school and environment metrics [i.e. length of school, depth of school, height of school, image compactness, the average amount of echo energy produced by the school per m² of lake surface (nautical area scattering coefficient (NASC)), lakebed depth, temperature, dissolved oxygen concentration, pH, turbidity, Chla concentration, and longitude]. This classifier showed a test classification accuracy of 85.4%. Evaluating the importance of each school metric showed that school length is the most important metric, followed by school height, school NASC, school depth, lakebed depth, and school image compactness. Environmental variables other than lake depth contribute very little to the overall classification performance and when all environmental information is removed, the overall RF accuracy is reduced by only ~1%.

For segmenting and classifying echo sounder data collected during acoustic trawl surveys, a DL-based method is presented by Brautaset *et al.* (2020). A slightly modified version of the U-Net architecture (Ronneberger *et al.*, 2015) is used as the classifier, which takes four frequency channels and a range-time subset of the echogram in the image format, resulting in the following classes: background, sandeel school, or other schools. The proposed method achieved significantly better results compared with non-DL methods when applied to a multifrequency dataset collected between 2007 and 2018 during the Norwegian sandeel survey.

Semmar and Vaz-dos-Santos (2019) present a simplex-based simulation approach developed to investigate growth regulation processes in fish populations, which was applied to *Merluccius hubbsi* stocks in the Southwestern Atlantic sampled in 1968–1972 and 2004 from six geographical areas. Using this approach, the

authors were able to show that the growth regulation of different body parts is related to the geographic origin of the fish. Liu *et al.* (2020) compared the performance of ensemble learning models—bagging trees (Johnson, 2001), RFs (Breiman, 2001), and boosting trees—using a dataset of 256 records of *Chondrichthyes* and *Osteichthyes* to predict fish natural mortality rate. The maximum age, growth coefficient, and asymptotic length were used as the features. The results show that tree-based ensemble learning models significantly improve the accuracy of fish natural mortality rate estimates compared with statistical regression models as well as the basic regression tree model (Breiman *et al.*, 1984). Among tested ensemble learning models, boosting trees and RFs performed the best, whereas the classification performance of boosting trees was slightly better.

Li *et al.* (2020) report on a publicly available dataset, PMID2019, containing 10 819 microscopic images of phytoplankton from 24 different categories. PMID2019 includes high resolution colour images with instance level annotations (manually labelled bounding boxes and corresponding species in each image) that can be used for phytoplankton detection. In order to generalize the dataset, Cycle-GAN (Zhu *et al.*, 2017) was applied to differentiate between images of dead and living cells so that images of dead and living cells can be inter-converted without losing their original features. This resulted in a synthetic phytoplankton living cell image dataset created from the original dead cell images that could be applied to detect phytoplankton *in situ*. PMID2019 was benchmarked by applying several state-of-the-art object detection algorithms: faster R-CNN (Ren *et al.*, 2015), feature pyramid network (Lin *et al.*, 2017a), single shot multiBox detector (Liu *et al.*, 2016), YOLOv3 (Redmon and Farhadi, 2018), and RetinaNet (Lin *et al.*, 2017b). Fast R-CNN produced the best results: average precision between 70.27 and 96.30% for different scenarios (e.g. various lighting conditions and complex background).

Campbell *et al.* (2020) present a novel plankton camera and propose a CNN-based classification system that was applied to the images collected. The plankton camera includes a 0.137 mm × 143 mm telecentric lens mounted on a 12-MP colour camera inside a large pressure housing with a sapphire glass optical port. The camera takes 12-bit colour images at a maximum frame rate of seven frames per second. This imager also incorporates an on-board computer system to segment each image and retain regions of interest that contain images of individual plankton using various image processing algorithms. The CNN architecture fine-tuned to classify the collected images was “Inception v3” (Szegedy *et al.*, 2015). The training set was composed of 18 868 images of 43 separate classes. Classification performance obtained on test data varied among the different classes.

ML and the future of marine science

The articles included in this themed set, and those that they cite make clear that there is great potential for ML to contribute to rapid advances in marine science. DL has already supported impressive advances by changing the way that experts analyse and interpret data, as well as in the amount of data that can be processed rapidly. However, the volume of data produced in marine science continues to increase and this introduces new challenges. Possible solutions follow.

- ML has to be more fully integrated, not only in processing marine data but also in the collection and management of data

and, therefore, ML scientists should collaborate more closely with marine scientists in data collection and infrastructure design.

- Communication between ML experts and marine scientists should be improved such that both sides become aware of the range of potential applications. There should be constant engagement between ML experts and marine biologists. ML experts should meet with stakeholders to develop and ensure a mutual understanding regarding the challenges of data analysis. On the other hand, marine experts should try to gain ML knowledge to better understand the potential and limitations of ML methods. This would serve to better define the desired accuracy of any ML pipeline.
- The transparency and intuitiveness of ML methods should be improved so that ML is more than a black box for marine scientists.
- Preserving and sharing ML knowledge and expertise within the marine science community: the size of marine data is huge, however, the size of data used to evaluate ML methods is generally very limited. This is because datasets contain an insufficient amount of labelled data. One solution could be establishing a common online repository in which researchers can share their data as well as their trained models and ML codes that would be aligned with their data.

We encourage submissions to this Journal that follow-up on these and related topics.

Acknowledgements

We are grateful to the authors for their enthusiasm and engagement in this initiative.

Funding

HIB's contribution to this article themed set was supported by Project # 83741 (Scientific publishing and editing) from the Institute of Marine Research, Norway.

References

- Aguzzi, J., Doya, C., Tecchio, S., De Leo, F. C., Azzurro, E., Costa, C., Sbragaglia, V., *et al.* 2015. Coastal observatories for monitoring of fish behaviour and their responses to environmental changes. *Reviews in Fish Biology and Fisheries*, 25: 463–483.
- Allken, V., Handegard, N. O., Rosen, S., Schreyeck, T., Mahiout, T., and Malde, K. 2019. Fish species identification using a convolutional neural network trained on synthetic data. *ICES Journal of Marine Science*, 76: 342–349.
- Álvarez-Ellacuría, A., Palmer, M., Catalán, I. A., and Lisani, J.-L. 2019. Image-based, unsupervised estimation of fish size from commercial landings using deep learning. *ICES Journal of Marine Science*, 77: 1330–1339.
- Aste, M., Boninsegna, M., Freno, A., and Trentin, E. 2015. Techniques for dealing with incomplete data: a tutorial and survey. *Pattern Analysis and Applications*, 18: 1–29.
- Badrinarayanan, V., Kendall, A., and Cipolla, R. 2017. SegNet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39: 2481–2495.
- Bartholomew, D. C., Mangel, J. C., Alfaro-Shigueto, J., Pingo, S., Jimenez, A., and Godley, B. J. 2018. Remote electronic monitoring as a potential alternative to on-board observers in small-scale fisheries. *Biological Conservation*, 219: 35–45.
- Beauchemin, S. S., and Barron, J. L. 1995. The computation of optical flow. *ACM Computing Surveys*, 27: 433–466.
- Beyan, C., Katsageorgiou, V., and Fisher, R. B. 2018. Extracting statistically significant behaviour from fish tracking data with and without large dataset cleaning. *IET Computer Vision*, 12: 162–170.
- Brautaset, O., Waldeland, A. U., Johnsen, E., Malde, K., Eikvil, L., Salberg, A.-B., and Handegard, N. O. 2020. Acoustic classification in multifrequency echosounder data using deep convolutional neural networks. *ICES Journal of Marine Science*, 77: 1391–1400.
- Breiman, L. 2001. Random forests. *Machine Learning*, 45: 5–32.
- Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. 1984. *Classification and Regression Trees*. Taylor & Francis, Belmont.
- Campbell, R. W., Roberts, P. L., and Jaffe, J. 2020. The Prince William Sound Plankton Camera: a profiling *in situ* observatory of plankton and particulates. *ICES Journal of Marine Science*, 77: 1440–1455.
- Cavazos, T., Comrie, A. C., and Liverman, D. M. 2002. Intraseasonal variability associated with wet monsoons in Southeast Arizona. *Journal of Climate*, 15: 2477–2490.
- Dargan, S., Kumar, M., Ayyagari, M. R., and Kumar, G. 2019. A survey of deep learning and its applications: a new paradigm to machine learning. *Archives of Computational Methods in Engineering*, doi: 10.1007/s11831-019-09344-w.
- De Robertis, A., Lawrence-Slavas, N., Jenkins, R., Wangen, I., Mordy, C. W., Meinig, C., Levine, M., *et al.* 2019. Long-term measurements of fish backscatter from Saildrone unmanned surface vehicles and comparison with observations from a noise-reduced research vessel. *ICES Journal of Marine Science*, 76: 2459–2470.
- Fan, J., Samworth, R., and Wu, Y. 2009. Ultrahigh dimensional feature selection: beyond the linear model. *The Journal of Machine Learning Research*, 10: 2013–2038.
- Fisher, R. B., Chen-Burger, Y.-H., Giordano, D., Hardman, L., and Lin, F.-P. 2016. *Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data*, 104. Springer, Cham.
- Frenay, B., and Verleysen, M. 2014. Classification in the presence of label noise: a survey. *IEEE Transactions on Neural Networks and Learning Systems*, 25: 845–869.
- Forget, G., Campin, J.-M., Heimbach, P., Hill, C. N., Ponte, R. M., and Wunsch, C. 2015. ECCO version 4: an integrated framework for non-linear inverse modeling and global ocean state estimation. *Geoscientific Model Development*, 8: 3071–3104.
- French, G., Mackiewicz, M., Fisher, M., Holah, H., Kilburn, R., Campbell, N., and Needle, C. 2019. Deep neural networks for analysis of fisheries surveillance video and automated monitoring of fish discards. *ICES Journal of Marine Science*, 77: 1340–1353.
- García, R., Prados, R., Quintana, J., Tempelaar, A., Gracias, N., Rosen, S., Vågstøl, H., *et al.* 2019. Automatic segmentation of fish using deep learning with application to fish size measurement. *ICES Journal of Marine Science*, 77: 1354–1366.
- Godø, O. R., Handegard, N. O., Browman, H. I., Macaulay, G. J., Kaartvedt, S., Giske, J., Ona, E., *et al.* 2014. Marine ecosystem acoustics (MEA): quantifying processes in the sea at the spatio-temporal scales on which they occur. *ICES Journal of Marine Science*, 71: 2357–2369.
- Goodfellow, I. J., Bulatov, Y., Ibarz, J., Arnoud, S., and Shet, V. 2014. Multi-digit number recognition from street view imagery using deep convolutional neural networks. *Proceedings of International Conference on Learning Representations (ICLR)*.
- He, K., Gkioxari, G., Dollar, P., and Girshick, R., 2017. Mask R-CNN. *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 1: 2980–2988.
- Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A.-r., Jaitly, N., Senior, A., *et al.* 2012. Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. *IEEE Signal Processing Magazine*, 29: 82–97.

- Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural Computation*, 9: 1735–1780.
- Huo, Y., Guan, J., Zhang, J., Zhang, M., Wen, J., and Lu, Z. 2019. Zero-shot learning with few seen class samples. *IEEE International Conference on Multimedia and Expo (ICME)*, 1: 1336–1341.
- James, S. C., Zhang, Y., and O'Donncha, F. 2018. To forecast wave conditions. *Coastal Engineering*, 137: 1–10.
- Janzen, C., McCammon, M., Weingartner, T., Statscewich, H., Winsor, P., Danielson, S., and Heim, R. 2019. Innovative real-time observing capabilities for remote coastal regions. *Frontiers in Marine Science*, 6: 176.
- Johnson, R. W. 2001. An introduction to the Bootstrap. *Teaching Statistics*, 23: 49–54.
- Joly, A., Goëau, H., Glotin, H., Spampinato, C., Bonnet, P., Vellinga, W.-P., Planqué, R., *et al.* 2015. LifeCLEF 2015: multimedia Life species identification challenges. *Proceedings of the 6th International Conference of the CLEF Association, CLEF'15*, 1: 462–483.
- Krasnopolsky, V. M. 2009. Neural Network Applications to Solve Forward and Inverse Problems in Atmospheric and Oceanic Satellite Remote Sensing, *Artificial Intelligence Methods in the Environmental Sciences*, pp. 191–205. Ed. by S. E. Haupt, A. Pasini, and C. Marzban. Springer, Dordrecht.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 1: 1097–1105.
- Kubat, M., Holte, R. C., and Matwin, S. 1998. Machine learning for the detection of oil spills in satellite radar images. *Machine Learning*, 30: 195–215.
- Li, Q., Sun, X., Dong, J., Song, S., Zhang, T., Liu, D., Zhang, H., *et al.* 2019. Developing a microscopic image dataset in support of intelligent phytoplankton detection using deep learning. *ICES Journal of Marine Science*, 77: 1427–1439.
- Lin, T. Y., Dollar, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. 2017a. Feature pyramid networks for object detection. *Proceedings of IEEE Conference on Computer Vision Pattern Recognition (CVPR)*, 42: 2117–2125.
- Lin, T. Y., Goyal, P., Girshick, R., He, K., and Dollar, P. 2017b. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2999–3007.
- Liu, C., Zhou, S., Wang, Y., and Hu, Z. 2020. Natural mortality estimation using tree-based ensemble learning methods. *ICES Journal of Marine Science*, 77: 1414–1426.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., and Berg, A. C. 2016. SSD: single shot MultiBox detector. *Proceedings of European Conference on Computer Vision (ECCV)*, 1: 21–37.
- Lombard, F., Boss, E., Waite, A. M., Vogt, M., Uitz, J., Stemmann, L., Sosik, H. M., *et al.* 2019. Globally consistent quantitative observations of planktonic ecosystems. *Frontiers in Marine Science*, 6: 196.
- Long, J., Shelhamer, E., and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 3431–3440.
- Lu, Y.-C., Tung, C. and Kuo, Y.-F. 2020. Identifying the species of harvested tuna and billfish using deep convolutional neural networks. *ICES Journal of Marine Science*, 77: 1318–1329.
- Lu, Y.-C., Tung, C., and Kuo, Y.-F. 2019. Detecting and counting harvested fish and identifying fish types in electronic monitoring system videos using deep convolutional neural networks. *ICES Journal of Marine Science*, 77: 1367–1378.
- Mahmood, A., Bennamoun, M., An, S., Sohel, F., Boussaid, F., Hovey, R., and Kendrick, G. 2019. Automatic detection of Western rock lobster using synthetic data. *ICES Journal of Marine Science*, 77: 1308–1317.
- Makarynskyy, O., Makarynska, D., Kuhn, M., and Featherstone, W. E. 2004. Predicting sea level variations with artificial neural networks at Hillarys Harbour, Western Australia Estuarine. *Coastal and Shelf Sciences*, 61: 351–360.
- Malde, K., Handegard, N. O., Eikvil, L., and Salberg, A.-B. 2019. Machine intelligence and the data-driven future of marine science. *ICES Journal of Marine Science*, 77: 1274–1285.
- McKnight, P. E., McKnight, K. M., Sidani, S., and Figueredo, A. J. 2007. *Missing Data: A Gentle Introduction*. Ed. by D. A. Kenny. The Guilford Press, New York. 251 pp.
- Partnership for Observation of the Global Ocean (POGO) Workshop on Machine Learning and Artificial Intelligence in Biological Oceanographic Observations. <http://ocean-partners.org/pogo-workshop-machine-learning-and-artificial-intelligence-biological-oceanographic-observations> (last accessed 17 November 2019).
- Probst, W. N. 2019. How emerging data technologies can increase the trust in fisheries. *ICES Journal of Marine Science*, 77: 1286–1294.
- Proud, R., Mangeni-Sande, R., Kayanda, R. J., Cox, M. J., Nyamweya, C., Ongore, C., Natugonza, V., *et al.* 2020. Automated classification of schools of the silver cyprinid *Rastrineobola argentea* in Lake Victoria acoustic survey data using random forests. *ICES Journal of Marine Science*, 77: 1379–1390.
- Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. 2016. You only look once: unified, real-time object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1: 779–788.
- Redmon, J., and Farhadi, A. 2018. ‘Yolov3: An incremental improvement’, *arXiv preprint arXiv: 1804.02767*, preprint: not peer reviewed.
- Ren, S., He, K., Girshick, R., and Sun, J. 2015. Faster R-CNN: towards real-time object detection with region proposal networks. *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 1: 91–99.
- Ronneberger, O., Fischer, P., and Brox, T. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation, *arXiv: 1505.04597 [cs.CV]*. <https://arxiv.org/abs/1505.04597> (last accessed 1 April 2020).
- Salman, A., Siddiqui, S. A., Shafait, F., Mian, A., Shortis, M., Khurshid, K., Ulges, A., *et al.* 2019. Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system. *ICES Journal of Marine Science*, 77: 1295–1307.
- Semmar, N., and Vaz-dos-Santos, A. M. 2019. Highlighting growth regulation processes in fish populations by a simplex simulation approach: application to *Merluccius hubbsi* stocks in the Southwestern Atlantic. *ICES Journal of Marine Science*, 77: 1401–1413.
- Siddiqui, S. A., Salman, A., Malik, M. I., Shafait, F., Mian, A., Shortis, M. R., and Harvey, E. S. 2018. Automatic fish species classification in underwater videos: exploiting pre-trained deep neural network models to compensate for limited labelled data. *ICES Journal of Marine Science*, 75: 374–389.
- Simonyan, K., and Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. *Proceedings of International Conference on Learning Representations (ICLR)*.
- Spampinato, C., Nadarajan, G., Chen-Burger, J., and Fisher, R. 2008. Detecting, tracking and counting fish in low quality unconstrained underwater videos. *Proceedings of 3rd International Conference on Computer Vision Theory and Applications (VISAPP)*, 2: 514–520.
- Stauffer, C., and Grimson, W. E. L. 1999. Adaptive background mixture models for real-time tracking. *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2: 246–252.
- Sutskever, I., Vinyals, O., and Le, Q. 2014. Sequence to sequence learning with neural networks. *Proceedings of Advances in Neural Information Processing Systems (NIPS)*, 27: 3104–3112.

- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. 2015. Rethinking the inception architecture for computer vision. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Tabak, M. A., Norouzzadeh, M. S., Wolfson, D. W., Sweeney, S. J., Vercauteren, K. C., Snow, N. P., Halseth, J. M., *et al.* 2019. Machine learning to classify animal species in camera trap images: applications in ecology. *Methods in Ecology and Evolution*, 10: 585–590.
- Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. 2014. DeepFace: closing the gap to human-level performance in face verification. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1: 1701–1708.
- Tanaka, A., Kishino, M., Doerffer, R., Schiller, H., Oishi, T., and Kubota, T. 2004. Development of a neural network algorithm for retrieving concentrations of chlorophyll, suspended matter and yellow substance from radiance data of the ocean color and temperature scanner. *Journal of Oceanography*, 60: 519–530.
- Thessen, A. 2016. Adoption of machine learning techniques in ecology and earth science. *One Ecosystem*, 1: e8621.
- Tseng, C.-H., and Kuo, Y.-F. 2020. Detecting and counting harvested fish and identifying fish types in electronic monitoring system videos using deep convolutional neural networks. *ICES Journal of Marine Science*, 77: 1367–1378.
- Van Engeland, T., Godø, O. R., Johnsen, E., Duineveld, G. C. A., and van Oevelen, D. 2019. Cabled ocean observatory data reveal food supply mechanisms to a coldwater coral reef. *Progress in Oceanography*, 172: 51–64.
- Verfuss, U. K., Aniceto, A. S., Harris, D. V., Gillespie, D., Fielding, S., Jiménez, G., Johnston, P., *et al.* 2019. A review of unmanned vehicles for the detection and monitoring of marine fauna. *Marine Pollution Bulletin*, 140: 17–29.
- Villon, S., Mouillot, D., Chaumont, M., Darling, E. S., Subsol, G., Claverie, T., and Villéger, S. 2018. A deep learning method for accurate and fast identification of coral reef fishes in underwater images. *Ecological Informatics*, 48: 238–244.
- Wu, A., Hsieh, W. W., and Tang, B. 2006. Neural network forecasts of the tropical Pacific sea surface temperatures. *Neural Networks*, 19: 145–154.
- Xiao, T., Xia, T., Yang, Y., Huang, C., and Wang, X. 2015. Learning from massive noisy labeled data for image classification. *Proceedings of Computer Vision and Pattern Recognition*, 1: 2691–2699.
- Zhu, J. Y., Park, T., Isola, P., and Efros, A. A. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, 1: 2223–2232.
- Zivkovic, Z., and Heijden, F. 2006. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters*, 27: 773–780.

Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Quo Vadimus

Machine intelligence and the data-driven future of marine science

Ketil Malde ^{1,2,*}, Nils Olav Handegard¹, Line Eikvil³, and Arnt-Børre Salberg³

¹Institute of Marine Research, Bergen, Norway

²Department of Informatics, University of Bergen, Norway

³Norwegian Computing Center, Oslo, Norway

*Corresponding author: tel: +47 98691834; e-mail: ketil.malde@imr.no.

Malde, K., Handegard, N. O., Eikvil, L., and Salberg, A-B. Machine intelligence and the data-driven future of marine science. – ICES Journal of Marine Science, 77: 1274–1285.

Received 12 September 2018; revised 7 February 2019; accepted 20 February 2019; advance access publication 15 April 2019.

Oceans constitute over 70% of the earth's surface, and the marine environment and ecosystems are central to many global challenges. Not only are the oceans an important source of food and other resources, but they also play a important roles in the earth's climate and provide crucial ecosystem services. To monitor the environment and ensure sustainable exploitation of marine resources, extensive data collection and analysis efforts form the backbone of management programmes on global, regional, or national levels. Technological advances in sensor technology, autonomous platforms, and information and communications technology now allow marine scientists to collect data in larger volumes than ever before. But our capacity for data analysis has not progressed comparably, and the growing discrepancy is becoming a major bottleneck for effective use of the available data, as well as an obstacle to scaling up data collection further. Recent years have seen rapid advances in the fields of artificial intelligence and machine learning, and in particular, so-called deep learning systems are now able to solve complex tasks that previously required human expertise. This technology is directly applicable to many important data analysis problems and it will provide tools that are needed to solve many complex challenges in marine science and resource management. Here we give a brief review of recent developments in deep learning, and highlight the many opportunities and challenges for effective adoption of this technology across the marine sciences.

Keywords: analysis bottleneck, convolutional neural networks, data processing, deep learning, observations, resource management.

Introduction

In March 2016, Google DeepMind pitched their computer programme AlphaGo (Silver *et al.*, 2016) against expert go player (ranked 9th dan) Lee Sedol in a five-game match, and won. This happened 20 years after IBM's chess playing computer *Deep Blue* famously played to a draw against grand master Gary Kasparov (Campbell *et al.*, 2002). Go is considered a notoriously difficult game for computers, and the event was widely reported in the press as an important milestone in the development of artificial intelligence (Wood, n.d.), and it was listed in *Science* as runner-up for the title of *Breakthrough of the Year* (Science, 2016).

Yet, this is only one of a series of remarkable achievements brought forth by recent developments in the field of artificial intelligence, and the triumph was soon overshadowed by new successes, for instance when AlphaZero managed to surpass human level skill in go, chess, and shogi solely from the experience it gathers playing against itself (Silver *et al.*, 2017).

Systems that are becoming increasingly intelligent are now being deployed on every scale, from mobile phones to supercomputers, and they are involved in a diversity of tasks, including personalized ranking of search results, selecting relevant advertisements, assisting vehicle driving, recognizing handwriting, and

understanding spoken commands. Common to these successes is the application of a new approach called *deep learning* (LeCun *et al.*, 2015).

Many of the high-profile uses of deep learning originate from corporations like Google, Facebook, Microsoft, and Amazon. These are consumer-oriented, technology-driven companies with access to large data repositories and computing resources (three of the four run commercial cloud services). Interestingly, these companies are also on the forefront of academic research, Google lists (Google, 2018) close to 1500 research papers on machine intelligence, perception, and translation, and another 380 on natural language processing. Microsoft reports publishing 239 papers on artificial intelligence in 2017 alone (Microsoft, 2018).

Technological progress has made data collection less costly, and this also affects the marine sciences. Large infrastructure projects are being developed to store and organize the data, and analysis is increasingly becoming a bottleneck. To meet many of the global challenges in marine science and management, it is necessary to realize the potential of collected data through automating more of the analysis. Here, we explore how new analysis technologies can be exploited to meet this goal.

Navigating an ocean of data

More than two-thirds of the planet is covered by oceans. The marine environment is a key component of the earth's climate, and its diverse ecosystems provide about half the global biological production and essential ecosystem services. The UN sustainability goals 2 (food security) and 3 (health) indirectly address the ocean, whereas goal 14 (use of oceans) explicitly acknowledges the need for sustainable development for the oceans and seas.

Marine science must rise to these challenges and provide the knowledge needed to ensure sustainable use of the marine environment. The necessity of an ecosystems approach to marine management is accepted worldwide (Pikitch *et al.*, 2004; Bianchi and Skjoldal, 2008; Koslow, 2009; Link and Browman, 2014) and is reflected in the [revised] European common fisheries policy and the marine strategy framework directive. Further development of models and observing systems is needed to meet these requirements, and a key challenge is how to extract relevant information when data volumes increases, data complexity increases, and data quality varies.

Increased data volumes

A direct consequence of improvements in sensor technology is an increase in data volume, usually accompanied by lower cost. This is brought about by several factors: higher data rates, decrease cost of sensor equipment, and for sensors operating *in situ*, advances in autonomous platforms technologies. New or upgraded sensors now allow us to observe essential ocean variables as well as other biological data, in both the field and the laboratory, at scales that were earlier beyond our ability. A few cases serve to illustrate this.

Acoustics is the primary sensor on acoustic-trawl surveys (MacLennan and Simmonds, 2005), and calibrated high-quality echo sounders are mounted on research vessels. These are now commonly installed on a wider range of platforms including vessels of opportunity, e.g. fishing vessels (Honkalehto *et al.*, 2011; Fassler *et al.*, 2016) and autonomous platforms, e.g. autonomous underwater vehicles (Fernandes *et al.*, 2003), gliders (Guihenet *et al.*, 2014), observatories (Godø *et al.*, 2014), and autonomous surface vehicles

(Mordy *et al.*, 2017). In concert these sensors could form an observation system that can inform ecosystem models (Handegard *et al.*, 2013), but the traditional manual data processing is a major bottleneck.

Research projects now routinely sequence the full genomes (Berthelot *et al.*, 2014; Lien *et al.*, 2016) or transcriptomes of tens or hundreds of individuals (Schunter *et al.*, 2014), resulting in several terabytes of data. Since the landmark Human Genome Project (Venter *et al.*, 2001), sequence costs have plummeted six orders of magnitude, and molecular methods are now used in new contexts like sequencing of marine communities to reveal its species composition or functional diversity (metagenomics) (Jackson *et al.*, 2015; Kodzius and Gojobori, 2015), or using genomic methods to investigate population structure, evolution, and migration patterns (Larson *et al.*, 2014; Malde *et al.*, 2017).

Camera equipment has become more advanced, robust, and inexpensive. Still and moving images are now used in a wide range of applications, including baited video surveys (Cappo *et al.*, 2007), benthic monitoring (Buhl-Mortensen *et al.*, 2015), in-trawl monitoring (Rosen *et al.*, 2013), plankton imaging (Stemann and Boss, 2012). Processing the resulting wealth of image data still often requires manual or partially manual labeling to extract meaningful information. In some cases, training data can be simulated (Figure 1), but often the lack of good training data hampers exploitation of technological advances and limits mass deployment of cameras.

Increased data complexity

Besides increased data quantity, new methods, and technology often let us collect and derive increasingly more complex data and information. This is true for model outputs and observations alike, and combining and analysing complex data are challenging



Figure 1. Simulated image mimicking output from the Deep Vision trawl camera solution. The simulator produces infinite training data for a classifier by producing random collages of fish images pasted onto an empty background. Image courtesy of Thomas Mahiout and Tiffanie Schreyeck.

since the relationships are often non-linear. Like for data quantity, the increased complexity applies almost universally, and a few cases are presented for illustration.

Early echo sounders emitted a single frequency, and received an intensity representing the reflected signal, conveniently plotted in a 2D diagram with time and depth (Sund, 1935). Multi-frequency equipment emits several frequencies simultaneously, and the difference in signal response provides valuable information about parameters like fish species, sizes, and orientations (Kloser *et al.*, 2002; Korneliusson and Ona, 2003). But the multiple diagrams are more demanding to interpret. Broadband equipment (Stanton *et al.*, 2010) replaces the multiple frequencies with continuous spectra, adding further complexity. Methods that can deal with these data have the potential to increase the information we get from the observations.

Similarly, most cameras capture visible light in the three primary colours corresponding to the photoreceptors in the human eye. In many cases, information is conveyed outside this spectrum, as evidenced by species like the mantis shrimps (*Stomatopoda* spp.), whose eyes have 16 different photoreceptors and the ability to detect both ultraviolet and polarized light (Marshall and Oberwinkler, 1999). Hyperspectral or multi-spectral photography that can record images both within and beyond the visible spectrum are likely to be useful in many settings, since light absorption and reflection of many substances strongly depend on the wavelength. For instance, the “colour” of the ocean is determined by the interactions of incident light with substances or particles present in the water. By exploiting multispectral data with fine spectral resolution several services provide frequent updates of a wide range of products based on the ocean colour (NASA, 2018). Methods to further exploit the increased data complexity are needed.

End to end ecosystem models have been proposed to be a key tool in integrated fisheries assessments (Fulton *et al.*, 2014). These models include components from physical forcing, geochemistry, primary production, and higher trophic levels, and the resulting model framework and model states are complex. Methods to extract relevant information, and often combining information from several sources are required, e.g. through ensemble modelling (Olsen *et al.*, 2016) or combining information from different data types. The state space from these models can be considered a complex data set and analysed as such. Methods to be able to find patterns and signals in the model states are needed.

Data quality

Improved technology generally leads to higher quality data, but occasionally increased data volumes are obtained by trading off quality for quantity. An example of this is research vessel surveys, which are costly to scale up. An alternative could be to collect data from the commercial fishing fleet, but with loss of rigid sampling design employed on research vessel surveys (Fassler *et al.*, 2016). Alternatively, relatively simple autonomous platforms could collect acoustics data, but without trawl sampling that has key information on age structure and species composition. Similarly, ARGO floats (Roemmich *et al.*, 2009) collect oceanographic data at a fraction of the cost of surveys using research vessels, but they can only drift with ocean currents, and we lose the ability to actively set up sampling designs or collect water samples. The information from increased data quantities may

compensate for a loss of quality, but the lack of rigid designs will often introduce biases which pose new challenges for analysis.

While the cases we highlight here exemplify the growing data volumes, increasing data complexity, and deteriorating data quality, they are not exhaustive. Rather they demonstrate how analysis increasingly is becoming a bottleneck for effective use of collected data across diverse fields and technologies. Relying on manual scrutiny by human experts does not scale well, and automatic analysis of data is necessary to alleviate a rapidly narrowing analysis bottleneck.

The deep learning revolution

Machine learning at a glance

A classical computer programme is an executable expression of an algorithm. That is, the programmer formulates a precise step-wise description of how to produce the desired result from the input. In contrast, a machine learning programme requires the programmer to specify only a more general model or architecture for the solution. The model is then *trained* using available data. Typically, training consists of gradually adjusting the parameters of the model, causing the programme to produce increasingly accurate results. By definition, a machine learning programme is a programme that is able to improve its performance from experience (Mitchell, 1997).

In principle, statistical methods like linear regression and estimation of probability distributions can be considered machine learning methods, but here we use the term to refer to more complex systems, like artificial neural networks, random forests, and support vector machines. And in contrast to statistical methods where the parameters are inherently meaningful, the parameters of more complex machine learning systems often capture some general pattern in the data in an opaque way, and the interpretation of the individual parameters can be difficult.

Neural networks

One of the archetypal machine learning systems, and a cornerstone of the recent revolution in machine learning, is the artificial neural network (Parker, 1985; Rumelhart *et al.*, 1986). It is conceptually simple, yet can solve complex problems, in fact, by the *universal approximation theorem* any function can be modelled by a neural network (Cybenko, 1989; Hornik *et al.*, 1989). A neural network consists of layers of simple computational units (or *neurons*), arranged so that the output of the units in one layer feed into the inputs of the next layer's units (Figure 2). Each unit calculates a weighted sum of its inputs, and applies a function (the *activation function*), $f(\cdot)$, that introduces non-linearity into the system. The weights, w_{ij} , of the inputs to each unit constitute the parameters to be learned. This is usually achieved using *back propagation* (Rumelhart *et al.*, 1986) to calculate the gradient for a cost function, which is then minimized iteratively using some variant of gradient descent.

Deep learning and the renaissance of neural networks

Work on neural networks in the 1980s and 1990s (Parker, 1985; Rumelhart *et al.*, 1986) was limited by computational power, lack of sufficiently large labelled data sets for training, and limitations in the learning algorithms. Hence, the dominant approach to machine learning was to use application dependent hand-designed features to describe the data in a compact form, reducing its dimensionality. For instance, computer vision would typically

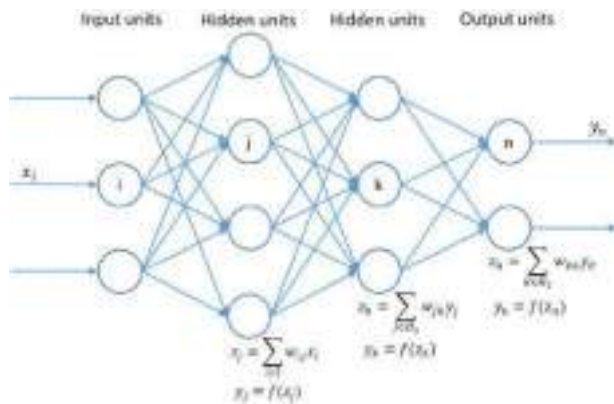


Figure 2. An artificial neural network typically consists of one input layer, several hidden layers, and one output layer. Each unit calculates a weighted sum of the inputs, and applies an activation function, f . For simplicity, we have omitted bias terms.

preprocess input images with a manually designed programme to detect features like edges and corners (Lowe, 2004; Dalal and Triggs, 2005). Classification algorithms like decision trees, shallow neural networks, and support vector machines (Boser *et al.*, 1992) would then be applied to learn patterns from the features, rather than from the raw image data. The input features would typically be hand-crafted to each problem, and standard feature sets like scale-invariant feature transform (Lowe, 1999) or histogram of oriented gradients (Dalal and Triggs, 2005) were developed to be reusable and applicable to many image classification problems. Although these approaches were successful for many applications, there is a necessary trade-off between generality and the specific task at hand, and reusable but static features cannot capture the inherent complexity of many objects, nor translate easily to non-image or higher-dimensional data.

In recent years, the availability of computational power from the use of graphics processing units (GPUs) (Chellapilla *et al.*, 2006; Bergstra *et al.*, 2010) and distributed computing (Dean *et al.*, 2012), large annotated data sets like ImageNet (Russakovsky *et al.*, 2015) as well as algorithmic improvements (Nair and Hinton, 2010; Hinton, Srivastava, *et al.*, 2012; Ioffe and Szegedy, 2015; He *et al.*, 2016) has allowed the construction of much larger and deeper neural networks than before. The added complexity allows a network to learn relevant features in the data automatically, which is a defining element of *deep learning* (LeCun *et al.*, 2015; Schmidhuber, 2015). For instance in computer vision, the lower layers in the network learn to recognize primitive, general features like edges and corners in an image. Higher layers learn to identify more abstract features as combinations of features (e.g. object parts formed by primitive features). Finally, the highest layers learn to identify abstract classes as combinations of object parts. This hierarchical structure of the deep convolutional neural networks (CNNs) thus naturally models the hierarchical composition of the objects to be recognized.

In contrast to feature-specific machine learning, where generality of features is at odds with specificity to the problem, deep learning is a generalized *approach* for developing the solution simultaneously with the problem-specific features. Neural network architectures still benefit when tailored to specific data types and problems, but the ability of deep networks to learn the primitive features directly from the raw data makes the technology directly applicable to a wide range of problems.

CNNs and computer vision

CNNs (Fukushima, 1988; LeCun *et al.*, 1999) are structured as stacks of filters, each recognizing increasingly abstract features in the data. This approach is very effective for many image analysis problems, where objects are often recognized independent of their location. The convolutional network applies the same set of filters to all parts of the image, recognizing the same kinds of features regardless of their position. This leads to a dramatic reduction in the number of weights and consequently a reduction in training effort and data requirement.

In 2012, Krizhevsky *et al.* (2012) demonstrated that deep convolutional networks could obtain substantially higher image classification accuracy on the ImageNet Large Visual Recognition Challenge (ILSVRC) (Russakovsky *et al.*, 2015) than competing systems. Their success was a result of designing a deep CNN and training it using new and more efficient strategies, including rectifying non-linearities (ReLUs) (Nair and Hinton, 2010; He *et al.*, 2015; Xu *et al.*, 2015), dropout regularization (Srivastava *et al.*, 2014), and batch normalization (Ioffe and Szegedy, 2015). To train a CNN with performance metrics comparable to the ones reported by Krizhevsky *et al.* (2012), a substantial amount of labelled training images is needed, in addition to sufficient computational power (e.g. parallel computers or GPU accelerators).

The great improvements demonstrated by Krizhevsky *et al.* (2012) were followed by a sequence of increasingly successful ILSVRC contestants using deep neural networks (Zeiler and Fergus, 2014; Badrinarayanan *et al.*, 2015; Long *et al.*, 2014; Yu and Koltun, 2015; He *et al.*, 2016), and have placed image recognition tasks at the centre of an ongoing deep learning revolution. Similar techniques have been extended to object localization by identifying their coordinates and bounding boxes (Ren *et al.*, 2015; Redmon *et al.*, 2016). Related tasks are semantic segmentation, where individual pixels are mapped to classes representing different objects (Badrinarayanan *et al.*, 2015; Long *et al.*, 2014; Yu and Koltun, 2015; Chen *et al.*, 2018), and instance segmentation, where each instance of an object is identified in addition to being segmented (He *et al.*, 2017).

These challenges are important in their own right, and also pave the way towards complete scene understanding, a core computer vision problem that is important for a number of applications, including autonomous driving, human-machine interaction (Baccouche *et al.*, 2011), earth observation (Kampffmeyer *et al.*, 2016; Maggiori *et al.*, 2017), image search engines (Wan *et al.*, 2014), to name a few.

Beyond images

In many cases, machines exceed human level accuracy, e.g. for optical character recognition (Goodfellow *et al.*, 2013), face verification (Taigman *et al.*, 2014), and recognition of specialized object categories, like different breeds of dogs or species of birds (Xiao *et al.*, 2014). Even text obfuscated for the specific purpose of distinguishing humans from computers (so-called *captchas*) are ironically deciphered more accurately by computers than by humans (Goodfellow *et al.*, 2013). Deep learning has led to rapid advances in many other areas beside computer vision, and it has successfully been applied to problems like speech recognition (Hinton, Deng, *et al.*, 2012), machine translation (Sutskever *et al.*, 2014; Zhang *et al.*, 2015), and financial applications (Heaton *et al.*, 2017). The technology is starting to be applied to data analysis in many sciences, including high energy physics

(Baldi *et al.*, 2014), drug activity prediction (Dahl *et al.*, 2014), and visual processing of microscope data to reconstruct 3D models of brain tissue.

Similar methods can also be applied to unsupervised learning, where data are unlabelled. One of the most popular unsupervised deep learning approaches is the autoencoder (Hinton and Salakhutdinov, 2006), which typically learns a representation (encoding) of the data, by training the network to ignore signal “noise” (Vincent *et al.*, 2010). Another promising direction for clustering is to learn representations and simultaneously discover cluster structure in unlabelled data by optimizing a discriminative loss function. The deep embedded clustering (DEC) (Xie *et al.*, 2016) approach represents, to the best of our knowledge, the state-of-the-art. The DEC is based on an optimization strategy in which a neural network is pre-trained by means of an autoencoder and then fine-tuned by jointly optimizing cluster centroids in output space and the underlying feature representation. The restricted Boltzmann machine (RBM) is a type of neural network that may be used to model probability distributions. RBM algorithms are used in several applications, including dimensionality reduction (Hinton and Salakhutdinov, 2006), clustering (Beyan *et al.*, 2018), and collaborative filtering (Salakhutdinov *et al.*, 2007). Within marine science, Beyan *et al.* (2017) proposed an effective outlier detection algorithm that is based on cluster cardinality. Here, clusters were obtained applying a mean-covariance RBM to group the data such that data points in the same group are more similar to each other than to those in other groups.

Neural networks are often referred to as *feed forward* network, as each layer forwards its output to the inputs of the next. In contrast, recurrent neural networks (RNNs) (Pineda, 1987) incorporate one or more backward links, forming a cyclic architecture. This allows the network to retain information about previous states, and RNNs are therefore often applied to time series data. RNNs have been used successfully to model language (Hochreiter and Schmidhuber, 1997). Deeper models and special memory units called long short-term memory have allowed RNNs to achieve state of the art performance in e.g. speech recognition (Graves *et al.*, 2013).

Machine learning in marine science

The growing data volumes, increased data complexity, and reduced data quality pose challenges for the marine science discipline, but at the same time recent advances in machine learning offer new possibilities of addressing them. Systems for automatic data analysis can be considered on several levels, from making manual work more efficient to novel analyses of complex and heterogeneous data.

Emulating basic human expertise

Machine learning systems are typically trained to emulate human curation, and thus a natural application is to use such systems to replace labor intensive steps in existing analysis pipelines. Reliance of manual curation is currently limiting effective data use, and automatic systems can reduce cost or increase throughput, for instance identifying fish species from images (Allken *et al.*, 2019; Siddiqui *et al.*, 2018; Villon *et al.*, 2018), fish trajectory estimation (Beyan *et al.*, 2018), or automatic age reading of otoliths (Moen *et al.*, 2018). The latter is perhaps of particular interest, as it demonstrates that a deep learning can obtain an accuracy comparable to human curators. This is in contrast to

Fisher and Hunter (2018), who reviewed traditional, shallow machine learning approaches and concluded that they provided no substantial advantage over human curation.

A fully automated system with accuracy comparable to a human curator is ideal, but more limited systems have also merit. The ability to sort out irrelevant data (e.g. frames with no objects of interest in them) can reduce manual work by orders of magnitude, and rudimentary classifiers with limited accuracy can reduce it further. As a bonus, with an automatic system taking care of tedious routine and trivial cases, the curation work remaining for the human expert is likely to be more interesting and rewarding.

In many cases, less than perfect accuracy may be sufficient. For instance, in cases where the sampling variance is large, a small bias may be acceptable if a larger number of observations can be exploited. Analysis of plankton images often have many and variable categories and be confounded by detritus and variation in visibility and lighting conditions, and machine learning methods are often used to guide or assist the human curator (Uusitalo *et al.*, 2016). Furthermore, where judgement of human experts varies, automated systems are consistent and can be duplicated as needed. They are likely to be cheaper and easier to deploy in hostile conditions. And although initial systems may have an unsatisfactory accuracy, technology improves over time. With improved systems, data can be reanalysed with little effort.

Advancing beyond the human expert

In many cases, overwhelming data volumes means that automatic systems are necessary for analysis. But for an increasing number of tasks, machine learning systems can surpass human experts in quality as well as quantity.

Some tasks that can be solved in principle are still too complex in practice, even for human experts. Analysis can be elusive when systems consist of many different factors which interact in many different ways, ecosystems being a typical example. We may have knowledge of each species involved, their migratory behaviour, predators, and prey relationships, reproductive biology, and so on, and a species can be isolated in the lab and its behaviour and responses studied. However, aggregating this information and deriving the behaviour of complex systems in the wild is challenging. Instead, we often rely on complex ecosystem models based on assumed interactions between the various components, and make inferences about the system from the model results (Fulton *et al.*, 2003). This assumes that we have successfully included the key processes in our model and that we have correctly parameterized them. A common critique is that we rely too much on the assumptions (Planque, 2016). Another, more parsimonious, approach is to use conventional statistical models to fit the data, but these models may be too simplistic since non-linear effects are difficult to handle. The deep learning approach may offer a third approach, where the analysis is still based on observed data, but the system is more capable detect and model non-linearities. However, it is prudent to note that the information that we can extract from the data is limited by the information content in the first place. Even so, deep learning methods may be able to tease out patterns the other methods fail to do.

Gaining new scientific insights

A common criticism of many machine learning methods is that the resulting model is opaque: although it can be shown

empirically to work, it is often not clear *how* the model works, or what knowledge the model captures. For instance, the learned parameters of a linear regression have clear interpretations as slope and intercept. In contrast, the individual weights in a trained neural network do not carry any obvious meaning and can have very different significance for different inputs. This is analogous to human knowledge. As observed by Polanyi (2009), many tasks require knowledge that we are unable to express explicitly. For instance, we can recognize a face instantly, yet we are at a loss for describing the exact process of doing so. In science the goal is often to *understand* a phenomenon. This is often achieved by exploring model dynamics, but is less transparent in typical deep learning models.

Despite this opacity, it is nevertheless possible to get a glimpse of the knowledge embedded in a machine learning system. For instance, convolutional layers in deep neural networks often recognize specific features of the input. By identifying regions of the data (parts of an image, say) where specific neurons are triggered, we can observe the feature recognized by that neuron (Montavon *et al.*, 2018). Such an approach could for instance reveal whether a system of automatic otolith reading (Moen *et al.*, 2018) is counting rings, or whether it is using other geometric features, like shape or size, and to what extent each feature is informative.

A slightly different method consists of feeding the network noise, and then using a variant of back propagation to amplify elements of the input data that cause a particular classification result (Erhan *et al.*, 2009). Several variations of this method have been developed (Bach *et al.*, 2015; Yosinski *et al.*, 2015), producing synthetic images that illustrate the type of features used by the network to identify a certain class. While recognizable, the resulting image is not necessarily representative for actual data.

Reproducibility of science and improved processes

Marine science and management advice for marine resources go hand in hand. A data processing pipeline for management, starting with data collection, going through various analyses and simulations, and ending with stock forecasts and management advice, are central to many marine science institutions. Currently, this process contains several interpretation steps, where a human expert must examine data to extract information for use as input to subsequent steps.

Automating these interpretation steps gives us several advantages. First, the whole process becomes deterministic and reproducible. Verifying the model output from the input data can be done by simply rerunning the pipeline, and this helps build confidence in the results. More importantly, it lets researchers experiment with the model, adjusting its parameters and inputs to discover how they affect the output, and let us quantify the consequences of changes. For instance, one can estimate the effect of reducing cruise activities in favour of less expensive floats or autonomous stations, or whether deployment of more advanced equipment is justifiable. This knowledge will be important for optimizing resource usage and reduce uncertainty in the results.

Heterogeneous data and integrative analysis

Ecosystems are complex networks of biological, chemical, and physical factors which also includes human activities. It is unclear to what extent such systems can be understood from a reductionist approach of isolating and studying each component. That a more holistic approach is necessary is a key tenet of

transdisciplinary science (Nicolescu, 2008). But multi- and interdisciplinary approaches could also benefit marine science to a larger extent. For instance, molecular methods could complement traditional surveys for detecting the presence of species (Foote *et al.*, 2012; Thomsen *et al.*, 2012), cameras can detect fragile species that are destroyed by more intrusive methods (Remsen *et al.*, 2004), and autonomous platforms (Mordy *et al.*, 2017) could augment data from more traditional surveys. Integrative approaches could collect data from multiple databases representing a variety of collection regimes and scientific disciplines and reanalyse these data in new ways to derive new information. Making data interoperable is a key step for effective integrative analysis, and several large efforts aim at providing centralized infrastructures and standardized organization for data collected by third parties.

An advantage of machine learning methods is their ability to work well with ambiguous data. Deep learning methods incorporate multiple levels of representation (LeCun *et al.*, 2015). Lower layers learn less abstract representations of the input, and these methods can therefore be applied directly to data without preprocessing (e.g. to images represented as pixel arrays, or free-form text), identifying and extract salient features automatically. In contrast to shallower systems which depend on hand-crafted features, the relevant structure and information content in the data is captured implicitly by the model. This has allowed e.g. natural language processing systems using deep learning methods to deal with ambiguities and imprecision in human languages. This robustness is not limited to language, and allows us to construct compound systems with the ability to deal usefully with existing data that may be incomplete, inconsistent, ambiguous, and weakly structured (Raghupathi and Raghupathi, 2014).

Of course, deep learning systems can also be applied to more abstract features, or to combinations of features and raw data. For instance, a popular task in computer vision is automatic image caption generation, where image features (extracted directly from raw data) is fed into a RNN that generates the appropriate caption describing the image content (Vinyals *et al.*, 2014)

Challenges

To realize the potential of automatic analysis, we need effective methods capable of handling the large amounts of data generated. Although successful projects that apply deep learning in the marine sciences exist (ICES, 2018), the technology has not yet seen widespread deployment, and several obstacles must be overcome for successful development and implementation.

Data availability in a form suitable for analysis

One obstacle is the lack of large and well-structured data sets suitable for training machine learning models. There is considerable third party interest in machine learning, and online competitions like (Kaggle, 2018) show that the availability of clearly defined problems and curated data sets attracts expertise and effort. Current efforts to aggregate data in central data servers and to standardize formats and metadata are steps in the right direction, but it is important that such efforts are developed in concert with intended analysis. In many cases, new methods for unsupervised or semi-supervised analysis of data need to be developed.

Perhaps the most common problem is the lack of adequate metadata (in this context referring to response variables, classes, annotations, or labels). Large volumes of raw data are collected

and stored, but the specific and detailed results from analysis are not systematically recorded (Harris *et al.*, 2010), leaving the data essentially unannotated. In other cases, annotation is available, but made in an *ad hoc* manner. So where one annotator might label a plankton image “copepod, large,” another might label it “large copepod.” Often classes are poorly defined and inconsistent, and do not make use of available standards. And even when both data and metadata are available, in some cases the link between them is unreliable.

For applications with sparsely labelled training data, the discovery of the deep CNNs’ ability to generalize and the usefulness of transfer learning have been a break-through (Razavian *et al.*, 2014; Yosinski *et al.*, 2014; Azizpour *et al.*, 2014). Transfer learning concerns the concept of transferring knowledge from one area to another (usually related) domain, and fully pre-trained nets trained from large databases with large label spaces (e.g. ImageNet) have shown good performance on several tasks (Razavian *et al.*, 2014). The transfer learning for CNNs is typically performed by either using the pre-trained network as a feature extractor (Razavian *et al.*, 2014). This approach was used by Siddiqui *et al.* (2018), who used trained a support vector machine to classify fish species based on the output from a pre-trained neural network. An alternative is to fine-tune the pre-trained network on the new target data. Fine-tuning can be restricted to higher layers, as the nets here tend to become more specific to details of the original labels. Less abstract features from lower layers are often useful for new tasks without modification (Azizpour *et al.*, 2014).

Most lines of work study and solve the problem of transfer learning within the same modality. The cross-modality transfer problem has received less attention, but approaches considering this typically rely on the existence of paired modalities (Gupta *et al.*, 2016) or shared label spaces, for example by hallucinating modalities during training time (Hoffman *et al.*, 2016; Kampffmeyer *et al.*, 2017), or jointly embedding or learning representations from multiple modalities into a shared feature space. The existence of shared label spaces or images paired with labelled natural images is, however, not the rule for applications involving non-standard data, which is often the case within marine science.

Anchoring projects in existing infrastructure and pipelines

The value of data is in its use, and for marine data to be useful, it must be analysed and the output used in science, for resource management, or by industry. With data sets available, methods can readily be developed, but without integration into existing processes, the impact is small or non-existent. To reap the benefits of new methods, it is crucial to involve the whole value chain, from data collection, to data storage and management, to analysis, and final use of the information. Projects must seek to involve existing stakeholders and have long-term implementation as a central goal, i.e. technology on its own has no merit in this context.

Remote electronic monitoring systems like AIS (which broadcasts position and other information) have been in use for some time, and can be analysed to identify vessel activities. Such systems have been used successfully for effective enforcement of fisheries policies and marine protected areas (McCauley *et al.*, 2016). Vessel monitoring systems can provide more detailed information, e.g. monitoring catch and bycatch from video surveillance

(Joo *et al.*, 2011; French *et al.*, 2015). Electronic monitoring will enable effective and more specific policies for sustainable operations, but depends on automatic analysis to be cost effective (van Helmond *et al.*, 2017), and stakeholder commitment is crucial for implementation (ICES, 2018).

Developing new expertise and methods

Since Krizhevsky *et al.* (2012), machine learning has seen a tremendous increase in interest. In particular, many large, data-oriented corporations, including Google, Facebook, Amazon, Microsoft, IBM, and Baidu, are aggressively recruiting people with machine learning expertise. The academic sector is struggling to compete with enterprises for competence, and recruitment of experienced academic personnel to the commercial sector is likely to impede development of solutions needed for scientific progress; as well as having negative consequences for the education and training that the commercial sector itself depends on.

Structures are needed that encourage development and retention of machine learning expertise in the marine sciences. There is a need to provide motivation and opportunities for people with this background to work closely with stakeholders in the marine domains. For standard problems like image classification, it may be sufficient to adopt methods from other fields, but when dealing with data types and problems that are more particular to marine sciences, interdisciplinary approaches are needed, and scientists need to understand both machine learning and the relevant disciplines like biology or oceanography.

Software tools and frameworks

Deep learning has proven to be an effective tool in many similar situations and fields, and several popular software packages now exist that can be downloaded, adapted, and deployed quickly and easily. TensorFlow (Abadi *et al.*, 2015) is a flexible framework that abstracts computing hardware, but which has a steep learning curve. Keras (Chollet *et al.*, 2015) builds on top of TensorFlow or Theano (Bergstra *et al.*, 2010), providing an easier to use, but less flexible interface. PyTorch (Paszke *et al.*, 2017) is another popular framework combining ease of use with expressive power. These frameworks are general and can be adapted to challenges in the marine domain with relative ease (Allken *et al.*, 2019; Moen *et al.*, 2018; Siddiqui *et al.*, 2018; Villon *et al.*, 2018). The vast number of online tutorials and documentation is a major advantage, and pre-trained models are available from public repositories (often referred to as *model zoos*). Although these are usually aimed at generic tasks like classification of standard image data sets, they accelerate development of specific solutions by providing well-tested architectures and initial parameters that are useful as a starting point (Orenstein and Beijbom, 2017) for further training.

Until recently, developing and applying advanced analysis methods required programming skills as well as a good understanding of methods and software frameworks. A variety of programming languages—Fortran, MatLab, C++, Java, and R, to name a few—are used in marine science, but the bulk of commercial and academic development of new machine learning methods targets Python. A lack of familiarity with Python could limit uptake of new technologies, or restrict developers to an inferior selection of tools and frameworks available in their preferred language.

We are also seeing the introduction of tools and libraries that target the marine sciences specifically. Such domain-specific solutions provide solutions that are tailored to common use cases and with intuitive interfaces. This can help to make the technology much more accessible for non-experts. One recent example is the VIAME toolkit (Dawkins *et al.*, 2017), which is an ambitious project that integrates data processing and analyses in a comprehensive framework, and supports multiple programming languages.

In conclusion there are several levels for which the user can use and deploy these techniques. In general, there is a trade-off between ease of use and flexibility, and choice of framework and methods must be tailored to the competence and ambitions of each individual project. The authors of this paper use Keras and Theano daily and have found they serve as a reasonable balance between flexibility and ease of use.

Conclusions

In the near future, the volume and complexity of marine data are expected to increase by orders of magnitude. Autonomous platforms already drift, float, sail, and glide across the ocean surface and below it, collecting large amounts of data at relatively low cost. Additional data are collected from commercial and other non-scientific vessels, and from stationary observatories. Simultaneously, sensor technology is advancing rapidly, increasing resolution and detail level of the collected information.

Deep learning and CNNs have made impressive advances, and is likely to change the way we interpret, analyse, and collect data. For classification or regression over large, regularly structured data, existing methods can be (and is) applied more or less directly. Similarly, methods exist that can deal with time series and textual data. More speculatively, techniques from deep learning aimed at dealing with large numbers of parameters may bring insights in how to better model complex adaptive systems.

Nevertheless, some moderation is warranted, and it is not sufficient merely to accumulate vast amounts of data and expect a clever enough algorithm to readily extract valuable insights. All data are not created equal, and no analysis will be able to extract information that is not present in the data. Careful design of surveys and experiments is and will remain important. Also, deep learning methods often perform well within its domain, but can give unpredictable results on unfamiliar data. When such methods are deployed, a regime of careful monitoring of performance and subsequent adjustments will be necessary.

The transition into a data rich science is a paradigm shift with important implications. Current sparse sampling regimes and population-based models can be replaced with comprehensive monitoring at high resolution, sometimes down to the individual level. For locations of particular interest, like rivers or spawning grounds, it is already within our reach to register the presence of each individual fish, and classifying its species as well as behaviour and interactions. But data collection on this scale requires data analysis capabilities well beyond current manual methods, and will only be realized when the analysis bottleneck is solved.

Acknowledgements

The authors would like to thank Robert Jenssen for valuable comments and discussion.

Funding

This work was supported by the COGMAR project, Research Council of Norway grant no 270966/O70, and by the Norwegian Ministry of Trade, Industry and Fisheries.

References

- Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S. *et al.* 2015. TensorFlow: Large-scale Machine Learning on Heterogeneous Systems. <https://www.tensorflow.org/> (last accessed 29 March 2019).
- Allken, V., Handegard, N. O., Rosen, S., Schreyeck, T., Mahiout, T., and Malde, K. 2019. Fish species identification using a convolutional neural network trained on synthetic data. *ICES Journal of Marine Science*, 76: 342–349.
- Azizpour, H., Razavian, A. S., Sullivan, J., Maki, A., and Carlsson, S. 2014. Factors of Transferability for a Generic ConvNet Representation. arXiv:1406.5774 [cs].
- Baccouche, M., Mamalet, F., Wolf, C., Garcia, C., and Baskurt, A. 2011. Sequential deep learning for human action recognition. *In International Workshop on Human Behavior Understanding*, pp. 29–39. Ed. by A. A. Salah and B. Lepri. Springer, Heidelberg.
- Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., and Samek, W. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS One*, 10: e0130140.
- Badrinarayanan, V., Kendall, A., and Cipolla, R. 2015. SegNet: a deep convolutional encoder-decoder architecture for image segmentation. arXiv, 1511.00561 [cs]. <http://arxiv.org/abs/1511.00561> (last accessed 31 August 2016). preprint: not peer reviewed.
- Baldi, P., Sadowski, P., and Whiteson, D. 2014. Searching for exotic particles in high-energy physics with deep learning. *Nature Communications*, 5: <http://www.nature.com/ncomms/2014/140702/ncomms5308/full/ncomms5308.html> (last accessed 4 September 2015).
- Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J. *et al.* 2010. Theano: a CPU and GPU math compiler in Python. *In Proc. 9th Python in Science Conf.*
- Berthelot, C., Brunet, F., Chalopin, D., Juanchich, A., Bernard, M., Noël, B., Bento, P. *et al.* 2014. The rainbow trout genome provides novel insights into evolution after whole-genome duplication in vertebrates. *Nature Communications*, 5: 3657.
- Beyan, C., Katsageorgiou, V.-M., and Fisher, R. B. 2018. Extracting statistically significant behaviour from fish tracking data with and without large dataset cleaning. *IET Computer Vision*, 12: 162–170.
- Bianchi, G., and Skjoldal, H. R. 2008. *The Ecosystem Approach to Fisheries*. CABI, Oxfordshire. 379 pp.
- Boser, B. E., Guyon, I. M., and Vapnik, V. N. 1992. A training algorithm for optimal margin classifiers. *In Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, pp. 144–152. Ed. by D. Haussler. ACM, New York, NY.
- Buhl-Mortensen, L., Buhl-Mortensen, P., Dolan, M. F. J., and Holte, B. 2015. The MAREANO programme—a full coverage mapping of the Norwegian off-shore benthic environment and fauna. *Marine Biology Research*, 11: 4–17.
- Campbell, M., Hoane, A. J., and Hsu, F. 2002. Deep Blue. *Artificial Intelligence*, 134: 57–83.
- Cappo, M., Harvey, E. S., and Shortis, M. R. 2007. Counting and measuring fish with baited video techniques—an overview. *In Australian Society for Fish Biology 2006 Workshop Proceedings*, pp. 101–114. <http://epubs.aims.gov.au/handle/11068/7468> (last accessed 14 February 2018).
- Chellapilla, K., Puri, S., and Simard, P. 2006. High Performance Convolutional Neural Networks for Document Processing. *In Tenth International Workshop on Frontiers in Handwriting Recognition*. Suvisoft.

- Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L. 2018. Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40: 834–848. [10.1109/TPAMI.2017.2699184]
- Chollet, F. *et al.* 2015. Keras. <https://keras.io> (last accessed 29 March 2019).
- Cybenko, G. 1989. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2: 303–314.
- Dahl, G. E., Jaitly, N., and Salakhutdinov, R. 2014. Multi-task neural networks for QSAR predictions. arXiv, 1406.1231 [cs, stat]. preprint: not peer reviewed.
- Dalal, N., and Triggs, B. 2005. Histograms of oriented gradients for human detection. *In* 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), pp. 886–893. IEEE.
- Dawkins, M., Sherrill, L., Fieldhouse, K., Hoogs, A., Richards, B., Zhang, D., Prasad, L. *et al.* 2017. An open-source platform for underwater image and video analytics. *In* 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), pp. 898–906. IEEE.
- Dean, J., Corrado, G., Monga, R., Chen, K., Devin, M., Mao, M., Ranzato, M. *et al.* 2012. Large scale distributed deep networks. *In* *Advances in Neural Information Processing Systems* 25, pp. 1223–1231. Ed. by F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger. Curran Associates, Inc., Red Hook. <http://papers.nips.cc/paper/4687-large-scale-distributed-deep-networks.pdf> (last accessed 4 September 2015).
- Erhan, D., Bengio, Y., Courville, A., and Vincent, P. 2009. Visualizing higher-layer features of a deep network. University of Montreal, 1341: 1.
- Fassler, S. M. M., Brunel, T., Gastauer, S., and Burggraaf, D. 2016. Acoustic data collected on pelagic fishing vessels throughout an annual cycle: operational framework, interpretation of observations, and future perspectives. *Fisheries Research*, 178: 39–46.
- Fernandes, P. G., Stevenson, P., Brierley, A. S., Armstrong, F., and Simmonds, E. J. 2003. Autonomous underwater vehicles: future platforms for fisheries acoustics. *ICES Journal of Marine Science: Journal du Conseil*, 60: 684–691.
- Fisher, M., and Hunter, E. 2018. Digital imaging techniques in otolith data capture, analysis and interpretation. *Marine Ecology Progress Series*, 598: 213–231.
- Foote, A. D., Thomsen, P. F., Sveegaard, S., Wahlberg, M., Kielgast, J., Kyhn, L. A., Salling, A. B. *et al.* 2012. Investigating the potential use of environmental DNA (eDNA) for genetic monitoring of marine mammals. *PLoS One*, 7: e41781.
- French, G., Fisher, M., Mackiewicz, M., and Needle, C. 2015. Convolutional Neural Networks for Counting Fish in Fisheries Surveillance Video. *In* *Proceedings of the Machine Vision of Animals and their Behaviour (MVAB)*, pp. 7.1–7.10. Ed. by S. Amaral, T. Matthews, T. Plötz, S. McKenna, and R. Fisher. BMVA Press, University of Swansea. <https://ueaeprints.uea.ac.uk/55574/> (last accessed 31 January 2019).
- Fukushima, K. 1988. Neocognitron: a hierarchical neural network capable of visual pattern recognition. *Neural Networks*, 1: 119–130.
- Fulton, E. A., Smith, A. D. M., and Johnson, C. R. 2003. Effect of complexity on marine ecosystem models. *Marine Ecology Progress Series*, 253: 1–16.
- Fulton, E. A., Smith, A. D. M., Smith, D. C., and Johnson, P. 2014. An integrated approach is needed for ecosystem based fisheries management: insights from ecosystem-level management strategy evaluation. *PLoS One*, 9: e84242.
- Godø, O. R., Johnsen, S., and Torkelsen, T. 2014. The LoVe ocean observatory is in operation. *Marine Technology Society Journal*, 48: 24–30.
- Goodfellow, I. J., Bulatov, Y., Ibarz, J., Arnoud, S., and Shet, V. 2013. Multi-digit number recognition from street view imagery using deep convolutional neural networks. arXiv, 1312.6082 [cs]. <http://arxiv.org/abs/1312.6082> (last accessed 4 September 2015). preprint: not peer reviewed.
- Google. 2018. Publications. <https://ai.google/research/pubs/> (last accessed 9 May 2018).
- Graves, A., Mohamed, A., and Hinton, G. 2013. Speech recognition with deep recurrent neural networks. *In* 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 6645–6649. IEEE.
- Guihen, D., Fielding, S., Murphy, E. J., Heywood, K. J., and Griffiths, G. 2014. An assessment of the use of ocean gliders to undertake acoustic measurements of zooplankton: the distribution and density of Antarctic krill (*Euphausia superba*) in the Weddell Sea. *Limnology and Oceanography: Methods*, 12: 373–389.
- Gupta, S., Hoffman, J., and Malik, J. 2016. Cross modal distillation for supervision transfer. *In* 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2827–2836. IEEE.
- Handegard, N. O., Buisson, L., du Brehmer, P., Chalmers, S. J., De Robertis, A., Huse, G., Kloser, R. *et al.* 2013. Towards an acoustic-based coupled observation and modelling system for monitoring and predicting ecosystem dynamics of the open ocean. *Fish and Fisheries*, 14: 605–615.
- Harris, G., Thompson, R., Childs, J. L., and Sanderson, J. G. 2010. Automatic storage and analysis of camera trap data. *The Bulletin of the Ecological Society of America*, 91: 352–360.
- He, K., Gkioxari, G., Dollár, P., and Girshick, R. 2017. Mask R-CNN. arXiv, 1703.06870 [cs]. preprint: not peer reviewed.
- He, K., Zhang, X., Ren, S., and Sun, J. 2016. Deep residual learning for image recognition. *In* *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778. IEEE.
- He, K., Zhang, X., Ren, S., and Sun, J. 2016. Deep residual learning for image recognition. *In* *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778. IEEE.
- Heaton, J. B., Polson, N. G., and Witte, J. H. 2017. Deep learning for finance: deep portfolios. *Applied Stochastic Models in Business and Industry*, 33: 3–12.
- Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A., Jaitly, N., Senior, A. *et al.* 2012. Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. *IEEE Signal Processing Magazine*, 29: 82–97.
- Hinton, G. E., and Salakhutdinov, R. R. 2006. Reducing the dimensionality of data with neural networks. *Science*, 313: 504–507.
- Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. R. 2012. Improving neural networks by preventing co-adaptation of feature detectors. arXiv:1207.0580 [cs]. preprint: not peer reviewed.
- Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural Computation*, 9: 1735–1780.
- Hoffman, J., Gupta, S., and Darrell, T. 2016. Learning with side information through modality hallucination. *In* 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 826–834. IEEE.
- Honkalehto, T., Ressler, P. H., Towler, R. H., and Wilson, C. D. 2011. Using acoustic data from fishing vessels to estimate walleye pollock (*Theragra chalcogramma*) abundance in the eastern Bering Sea. *Canadian Journal of Fisheries and Aquatic Sciences*, 68: 1231–1242.
- Hornik, K., Stinchcombe, M., and White, H. 1989. Multilayer feed-forward networks are universal approximators. *Neural Networks*, 2: 359–366.
- ICES. 2018. Report of the Workshop on Machine Learning in Marine Science (WKMLEARN), 16–20 April 2018, ICES Headquarters, Copenhagen, Denmark. ICES CM 2018/EOSG: 28 pp.
- Ioffe, S., and Szegedy, C. 2015. Batch normalization: accelerating deep network training by reducing internal covariate shift. arXiv: 1502.03167 [cs]. preprint: not peer reviewed.
- Jackson, S. A., Borchert, E., O’Gara, F., and Dobson, A. D. 2015. Metagenomics for the discovery of novel biosurfactants of

- environmental interest from marine ecosystems. *Current Opinion in Biotechnology*, 33: 176–182.
- Joo, R., Bertrand, S., Chaigneau, A., and Ñiquen, M. 2011. Optimization of an artificial neural network for identifying fishing set positions from VMS data: an example from the Peruvian anchovy purse seine fishery. *Ecological Modelling*, 222: 1048–1059.
- Kaggle. 2018. <https://www.kaggle.com/> (last accessed 11 July 2018).
- Kampffmeyer, M., Salberg, A.-B., and Jenssen, R. 2016. Semantic segmentation of small objects and modeling of uncertainty in urban remote sensing images using deep convolutional neural networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 1–9. IEEE. http://www.cv-foundation.org/openaccess/content_cvpr_2016_workshops/w19/html/Kampffmeyer_Semantic_Segmentation_of_CVPR_2016_paper.html (last accessed 31 August 2016).
- Kampffmeyer, M., Salberg, A.-B., and Jenssen, R. 2017. Urban land cover classification with missing data using deep convolutional neural networks. *In 2017 IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 5161–5164. IEEE.
- Kloser, R. J., Ryan, T., Sakov, P., Williams, A., and Koslow, J. A. 2002. Species identification in deep water using multiple acoustic frequencies. *Canadian Journal of Fisheries and Aquatic Sciences*, 59: 1065–1077.
- Kodzius, R., and Gojoberi, T. 2015. Marine metagenomics as a source for bioprospecting. *Marine Genomics*, 24: 21–30.
- Korneliussen, R. J., and Ona, E. 2003. Synthetic echograms generated from the relative frequency response. *ICES Journal of Marine Science: Journal du Conseil*, 60: 636–640.
- Koslow, J. A. 2009. The role of acoustics in ecosystem-based fishery management. *ICES Journal of Marine Science*, 66: 966–973.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. 2012. ImageNet classification with deep convolutional neural networks. *In Advances in Neural Information Processing Systems 25*, pp. 1097–1105. Ed. by F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger. Curran Associates, Inc., Red Hook. <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf> (last accessed 4 September 2015).
- Larson, W. A., Seeb, L. W., Everett, M. V., Waples, R. K., Templin, W. D., and Seeb, J. E. 2014. Genotyping by sequencing resolves shallow population structure to inform conservation of Chinook salmon (*Oncorhynchus tshawytscha*). *Evolutionary Applications*, 7: 355–369.
- LeCun, Y., Bengio, Y., and Hinton, G. 2015. Deep learning. *Nature*, 521: 436–444.
- LeCun, Y., Haffner, P., Bottou, L., and Bengio, Y. 1999. Object recognition with gradient-based learning. *In Shape, Contour and Grouping in Computer Vision*, pp. 319–345. Ed. by D. A. Forsyth, J. L. Mundy, V. di Gesù, and R. Cipolla. Springer, Berlin, Heidelberg. https://link.springer.com/chapter/10.1007/3-540-46805-6_19 (last accessed 10 July 2018).
- Lien, S., Koop, B. F., Sandve, S. R., Miller, J. R., Kent, M. P., Nome, T., Hvidsten, T. R. *et al.* 2016. The Atlantic salmon genome provides insights into rediploidization. *Nature*, 533: 200–205.
- Link, J., and Browman, H. 2014. Integrating what? Levels of marine ecosystem-based assessment and management. *ICES Journal of Marine Science*, 71: 1170–1173.
- Long, J., Shelhamer, E., and Darrell, T. 2014. Fully convolutional networks for semantic segmentation. *arXiv:1411.4038 [cs]*.
- Lowe, D. G. 1999. Object recognition from local scale-invariant features. *In Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, pp. 1150–1157. IEEE.
- Lowe, D. G. 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60: 91–110.
- MacLennan, D., and Simmonds, E. J. 2005. *Fisheries Acoustics. Fish and Aquatic Resources Series 10*. Chapman & Hall, London.
- Maggiori, E., Tarabalka, Y., Charpiat, G., and Alliez, P. 2017. Convolutional neural networks for large-scale remote-sensing image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 55: 645–657. [10.1109/TGRS.2016.2612821]
- Malde, K., Seliussen, B. B., Quintela, M., Dahle, G., Besnier, F., Skaug, H. J., Øien, N. *et al.* 2017. Whole genome resequencing reveals diagnostic markers for investigating global migration and hybridization between minke whale species. *BMC Genomics*, 18: 76.
- Marshall, J., and Oberwinkler, J. 1999. Ultraviolet vision: the colourful world of the mantis shrimp. *Nature*, 401: 873–874.
- McCauley, D. J., Woods, P., Sullivan, B., Bergman, B., Jablonicky, C., Roan, A., Hirshfield, M. *et al.* 2016. Ending hide and seek at sea. *Science*, 351: 1148–1150.
- Microsoft. 2018. Search. <https://www.microsoft.com/en-us/research/search/> (last accessed 10 May 2018).
- Mitchell, T. M. 1997. *Machine Learning*. WCB. McGraw-Hill, Boston, MA.
- Moen, E., Handegard, N. O., Allken, V., Albert, O. T., Harbitz, A., and Malde, K. 2018. Automatic interpretation of otoliths using deep learning. *PLoS One*, 13: e0204713.
- Montavon, G., Samek, W., and Müller, K.-R. 2018. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73: 1–15.
- Mordy, C. W., Cokelet, E. D., De Robertis, A., Jenkins, R., Kuhn, C. E., Lawrence-Slavas, N., Berchok, C. L. *et al.* 2017. Advances in ecosystem research: Saldrome surveys of oceanography, fish, and marine mammals in the Bering Sea. *Oceanography*, 30: 113–115.
- Nair, V., and Hinton, G. E. 2010. Rectified linear units improve restricted Boltzmann machines. *In Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pp. 807–814. Omnipress.
- NASA. 2018. NASA Ocean Color. <https://oceancolor.gsfc.nasa.gov/> (last accessed 10 July 2018).
- Nicolescu, B. 2008. *Transdisciplinarity: Theory and Practice*. Hampton Press, New York. 332 pp.
- Olsen, E., Fay, G., Gaichas, S., Gamble, R., Lucey, S., and Link, J. S. 2016. Ecosystem model skill assessment. *Yes We Can! PLoS One*, 11: e0146467.
- Orenstein, E. C., and Beijbom, O. 2017. Transfer learning and deep feature extraction for planktonic image data sets. *In 2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1082–1088. IEEE.
- Parker, D. B. 1985. *Learning-logic: learning-logic: casting the cortex of the human brain in silicon*. Technical Report Tr-47, Center for Computational Research in Economics and Management Science. MIT Cambridge, MA.
- Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z. *et al.* 2017. Automatic differentiation in PyTorch. <https://openreview.net/forum?id=BJjrmfCZ> (last accessed 20 November 2018).
- Pikitch, E. K., Santora, C., Babcock, E. A., Bakun, A., Bonfil, R., Conover, D. O., Dayton, P. *et al.* 2004. Ecosystem-based fishery management. *Science*, 305: 346–347.
- Pineda, F. J. 1987. Generalization of back-propagation to recurrent neural networks. *Physical Review Letters*, 59: 2229–2232.
- Planque, B. 2016. Projecting the future state of marine ecosystems, “la grande illusion”? *ICES Journal of Marine Science*, 73: 204–208.
- Polanyi, M. 2009. *The Tacit Dimension*. University of Chicago Press, Chicago. 129 pp.
- Raghupathi, W., and Raghupathi, V. 2014. Big data analytics in healthcare: promise and potential. *Health Information Science and Systems*, 2: 3.
- Razavian, A. S., Azizpour, H., Sullivan, J., and Carlsson, S. 2014. CNN features off-the-shelf: an astounding baseline for

- recognition. *In* Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 512–519. IEEE Computer Society, Washington, DC. <http://dx.doi.org/10.1109/CVPRW.2014.131> (last accessed 14 February 2018).
- Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. 2016. You only look once: unified, real-time object detection. *In* Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788. IEEE.
- Remsen, A., Hopkins, T. L., and Samson, S. 2004. What you see is not what you catch: a comparison of concurrently collected net, Optical Plankton Counter, and Shadowed Image Particle Profiling Evaluation Recorder data from the northeast Gulf of Mexico. *Deep Sea Research Part I: Oceanographic Research Papers*, 51: 129–151.
- Ren, S., He, K., Girshick, R., and Sun, J. 2015. Faster R-CNN: towards real-time object detection with region proposal networks. arXiv: 1506.01497 [cs].
- Roemmich, D., Johnson, G. C., Riser, S., Davis, R., Gilson, J., Owens, W. B., Garzoli, S. L. *et al.* 2009. The Argo Program: observing the global ocean with profiling floats. *Oceanography*, 22: 34–43.
- Rosen, S., Jørgensen, T., Hammersland-White, D., and Holst, J. C. 2013. DeepVision: a stereo camera system provides highly accurate counts and lengths of fish passing inside a trawl. *Canadian Journal of Fisheries and Aquatic Sciences*, 70: 1456–1467.
- Rumelhart, D. E., Hinton, G. E., and Williams, R. J. 1986. Learning representations by back-propagating errors. *Nature*, 323: 533–536.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z. *et al.* 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115: 211–252.
- Salakhutdinov, R., Mnih, A., and Hinton, G. 2007. Restricted Boltzmann machines for collaborative filtering. *In* Proceedings of the 24th international conference on Machine learning, ACM, pp. 791–798.
- Schmidhuber, J. 2015. Deep learning in neural networks: an overview. *Neural Networks*, 61: 85–117.
- Schunter, C., Vollmer, S. V., Macpherson, E., and Pascual, M. 2014. Transcriptome analyses and differential gene expression in a non-model fish species with alternative mating tactics. *BMC Genomics*, 15: 167.
- Science. 2016. From AI to Protein Folding: Our Breakthrough Runners-up. <http://www.sciencemag.org/news/2016/12/ai-protein-folding-our-breakthrough-runners> (last accessed 9 May 2018).
- Siddiqui, S. A., Salman, A., Malik, M. I., Shafait, F., Mian, A., Shortis, M. R., and Harvey, E. S. 2018. Automatic fish species classification in underwater videos: exploiting pre-trained deep neural network models to compensate for limited labelled data. *ICES Journal of Marine Science*, 75: 374–389.
- Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Driessche, G., van den Schrieffwieser, J. *et al.* 2016. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529: 484–489.
- Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T. *et al.* 2017. Mastering the game of Go without human knowledge. *Nature*, 550: 354–359.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15: 1929–1958.
- Stanton, T. K., Chu, D., Jech, J. M., and Irish, J. D. 2010. New broadband methods for resonance classification and high-resolution imagery of fish with swimbladders using a modified commercial broadband echosounder. *ICES Journal of Marine Science: Journal du Conseil*, 67: 365–378.
- Stemmann, L., and Boss, E. 2012. Plankton and particle size and packaging: from determining optical properties to driving the biological pump. *Annual Review of Marine Science*, 4: 263–290.
- Sund, O. 1935. Echo sounding in fishery research. *Nature*, 135: 953.
- Sutskever, I., Vinyals, O., and Le, Q. V. V. 2014. Sequence to sequence learning with neural networks. *In* Advances in Neural Information Processing Systems 27, pp. 3104–3112. Ed. by Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger. Curran Associates, Inc., Red Hook. <http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf> (last accessed 4 September 2015).
- Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. 2014. DeepFace: closing the gap to human-level performance in face verification. *In* 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1701–1708. IEEE.
- Thomsen, P. F., Kielgast, J., Iversen, L. L., Møller, P. R., Rasmussen, M., and Willerslev, E. 2012. Detection of a diverse marine fish fauna using environmental DNA from seawater samples. *PLoS One*, 7: e41732.
- Uusitalo, L., Fernandes, J. A., Bachiller, E., Tasala, S., and Lehtiniemi, M. 2016. Semi-automated classification method addressing marine strategy framework directive (MSFD) zooplankton indicators. *Ecological Indicators*, 71: 398–405.
- van Helmond, A. T. M., Chen, C., and Poos, J. J. 2017. Using electronic monitoring to record catches of sole (*Solea solea*) in a bottom trawl fishery. *ICES Journal of Marine Science*, 74: 1421–1427.
- Venter, J. C., Adams, M. D., Myers, E. W., Li, P. W., Mural, R. J., Sutton, G. G., Smith, H. O. *et al.* 2001. The sequence of the human genome. *Science* (New York, NY), 291: 1304–1351.
- Villon, S., Mouillot, D., Chaumont, M., Darling, E. S., Subsol, G., Claverie, T., and Villéger, S. 2018. A deep learning method for accurate and fast identification of coral reef fishes in underwater images. *Ecological Informatics*, 48: 238–244.
- Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., and Manzagol, P.-A. 2010. Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. *The Journal of Machine Learning Research*, 11: 3371–3408.
- Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. 2014. Show and tell: a neural image caption generator. arXiv:1411.4555 [cs]. preprint: not peer reviewed.
- Wan, L., Zeiler, M., Zhang, S., Cun, Y. L., and Fergus, R. 2013. Regularization of Neural Networks using DropConnect. *In* PMLR, pp. 1058–1066. <http://proceedings.mlr.press/v28/wan13.html> (last accessed 14 February 2018).
- Wood, G. 2018. Google’s AI Wins Fifth and Final Game Against Go Genius Lee Sedol. <https://www.wired.com/2016/03/googles-ai-wins-fifth-final-game-go-genius-lee-sedol/> (last accessed 8 May 2018).
- Xiao, T., Xu, Y., Yang, K., Zhang, J., Peng, Y., and Zhang, Z. 2014. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. arXiv:1411.6447 [cs].
- Xie, J., Girshick, R., and Farhadi, A. 2016. Unsupervised deep embedding for clustering analysis. *In* Proceedings of the 33rd International Conference on International Conference on Machine Learning, 48, pp. 478–487. Ed. by M. F. Balcan and K. Q. Weinberger. JMLR.org, New York, NY. <http://dl.acm.org/citation.cfm?id=3045390.3045442> (last accessed 31 January 2019).
- Xu, B., Wang, N., Chen, T., and Li, M. 2015. Empirical evaluation of rectified activations in convolutional network. arXiv:1505.00853 [cs, stat]. preprint: not peer reviewed.
- Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. 2014. How transferable are features in deep neural networks? *In* Advances in Neural Information Processing Systems 27, pp. 3320–3328. Ed. by Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q.

- Weinberger. Curran Associates, Inc., Red Hook. <http://papers.nips.cc/paper/5347-how-transferable-are-features-in-deep-neural-networks.pdf> (last accessed 14 February 2018).
- Yosinski, J., Clune, J., Nguyen, A., Fuchs, T., and Lipson, H. 2015. Understanding neural networks through deep visualization. arXiv: 1506.06579 [cs]. <http://arxiv.org/abs/1506.06579> (last accessed 27 November 2018). preprint: not peer reviewed.
- Yu, F., and Koltun, V. 2015. Multi-scale context aggregation by dilated convolutions. arXiv:1511.07122 [cs]. preprint: not peer reviewed.
- Zeiler, M. D., and Fergus, R. 2014. Visualizing and understanding convolutional networks. *In* European Conference on Computer Vision, pp. 818–833. Springer.
- Zhang, W., Li, R., Deng, H., Wang, L., Lin, W., Ji, S., and Shen, D. 2015. Deep convolutional neural networks for multi-modality iso-intense infant brain image segmentation. *NeuroImage*, 108: 214–224.

Handling editor: David Demer

Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Quo Vadimus

How emerging data technologies can increase trust and transparency in fisheries

Wolfgang Nikolaus Probst  *

Johann Heinrich von Thünen-Institute of Sea Fisheries, Herwigstraße 31, 27572 Bremerhaven, Germany

*Corresponding author: tel: +49 471 94460380; e-mail: nikolaus.probst@thuenen.de.

Probst, W. N. How emerging data technologies can increase trust and transparency in fisheries. – ICES Journal of Marine Science, 77: 1286–1294.

Received 11 October 2018; revised 14 February 2019; accepted 18 February 2019; advance access publication 14 March 2019.

The ubiquitous spread of digital networks has created techniques which can organize, store, and analyse large data volumes in an automated and self-administered manner in real time. These technologies will have profound impacts on policy, administration, economy, trade, society, and science. This article sketches how three digital data technologies, namely the blockchain, data mining, and artificial intelligence could impact commercial fisheries including producers, wholesalers, retailers, consumers, management authorities, and scientist. Each of these three technologies is currently experiencing an enormous boost in technological development and real-world implementation and is predicted to increasingly affect many aspects of fisheries and seafood trade. As any economic sector acting on global scales, fishing and seafood production are often challenged with a lack of trust along various steps of the production process and supply chain. Consumers are often not well informed on the origin and production methods of their product, management authorities can only partly control fishing and trading activities and producers can be challenged by low market prices and competition with peers. The emerging data technologies can improve the trust among agents within the fisheries sector by increasing transparency and availability of information from net to plate.

Keywords: artificial intelligence, blockchain, data mining, enforcement, supply chain, traceability

Introduction

Smartphones influence our lives through their multi-purpose versatility. They allow us to communicate, orientate, inform, educate, and shop at almost any location at any time. Several users touch their smartphone more than hundred times per day, and excessive smartphone use can even lead to mental addictions and psychological disorders (Kwon *et al.*, 2013; Haug *et al.*, 2015; Samaha and Hawi, 2016). Our reliance on smartphones indicates how ubiquitous the use of digital networking technologies have become in everyday life. Currently, a large proportion of digital innovation is focusing on gaining extra information out of the large volumes of exchanged data. The economic aim of this data gathering is to turn the inherent information into added value by gaining better insight into the behaviour of consumers and their demands, but also to make production and trading processes

more efficient by monitoring the flow of products, goods, and services. Digital integration is advancing in many economic sectors and thus it is only reasonable to assume that it will also affect fisheries production and associated economic branches.

This article sketches potential impacts of emerging internet-based data techniques, i.e. the blockchain, real-time data mining, and artificial intelligence (AI), and shows how these techniques can influence the way fisheries are operated and managed and fish products are traded and consumed. The article will focus on these three technologies, because they are based on gathering, organizing, recording, processing, and analysing large volumes of data and are expected to drive major technological, economic, and social developments in the future (Li *et al.*, 2015; Swan, 2015; Tapscott and Tapscott, 2016; Cath *et al.*, 2018). While data mining and AI are not new disciplines by themselves, they have

gained renewed interest by big-data applications and digital innovations, such as autonomous driving or image recognition (Rajamaran, 2016; Cath *et al.*, 2018).

Fisheries: a global and elusive enterprise

Fish and seafood products are harvested and traded worldwide (FAO, 2016). For many of these products, it is a long way from net to plate, as their supply chains cross multiple continents, national borders, trade zones, and jurisdictions. For consumers and national controlling agencies, it is therefore not easy to recognize the true origin of imported seafood products. However, even products of regional fisheries can be associated with knowledge gaps on catches, by-catches, and environmental impacts (Hall, 1996; Lewison *et al.*, 2004; Kaiser *et al.*, 2006; Benoit and Allard, 2009; Herr *et al.*, 2009). Consumers, producers, and management authorities are therefore faced with the challenge to place trust into producers and the members of the supply chain, while knowledge on the exact behaviour of the other participants is limited (Mosler, 1993).

In many commercial fisheries, in which the fishing companies are referred to as “producers”, only a proportion of the full catch (or capture) is retained and landed (Hall, 1996). At harbour, the landed catch is reported to wholesalers and management authorities before being traded and processed to be finally sold to the consumers. What really happens on board of fishing vessels, i.e. which species in which quantities are really caught, remains elusive to management authorities for the majority of fishing operations (Benoit and Allard, 2009; Edgar *et al.*, 2016).

Along the supply chain, various mechanism can operate to control the compliance of the supply chain members to regional and national jurisdictions. At sea, the national coast guard, observers-at-sea, or video-monitoring systems can inspect fishing practices or record the catch (Ulleweit *et al.*, 2010; Kindt-Larsen *et al.*, 2011; Haskell *et al.*, 2014), but the frequency of these control mechanisms is often negligible compared to the frequency of total fishing operations. On land, the most important control measure is the declaration of the landings to national authorities to allow the comparison against the allocated annual catch quota. Depending on the world region of landing, this mechanism covers a smaller or larger proportion of the landings, as not all landed species are quota restricted. Also at land, the production and processing of seafood products are controlled by governmental or self-committed industry institutions (consumer or eco-labels) (Gulbrandsen, 2009).

During all steps of the supply chain, the mechanisms of control can be cheated, leading to unreported and/or illegal catches (Helyar *et al.*, 2014). During the trading and processing chain, landings can also be re-declared into another, more valuable species or being caught from a different origin (stock or catch area). Increasing the traceability of fish and seafood products for consumers and management authorities has therefore become a designated aim of retailers and policy makers (Schröder, 2008).

Data technologies on the rise

The lack of transparency in fisheries production and trading processes has led to a trust crisis by consumers into management authorities and the industry (Jacquet and Pauly, 2007; Helyar *et al.*, 2014). Emerging data technologies may help to overcome some aspects of this crisis by improving the transparency for controlling agents such as end-consumers, NGOs, and management authorities.

Blockchain and smart contracts

Blockchains have become prominent through the boom of crypto currencies, for which blockchains provide the technical foundation. Since then blockchains have been modified in multiple way leading to an ever increasing number of blockchain projects and crypto currency tokens. But blockchains are not only used for crypto-currencies. In fact, many companies and independent institutions are currently looking for possible implementations of blockchains into their operations (Swan, 2015; Xu *et al.*, 2016, see also the IBM website for examples of blockchain applications).

In October 2008, an anonymous author with the pseudonym Satoshi Nakamoto released a whitepaper for a non-institutional (i.e. non-governmental) currency system called Bitcoin. Bitcoin is based on a blockchain that records financial transactions in a decentralized database. Thereby, Bitcoin was originally perceived by a community of enthusiast as an alternative and independent financial system, which could overcome institutional structures which lead to the financial crisis of 2008. The blockchain database, generally referred to as distributed ledger, adds new entries chronologically within a blockchain network (Swan, 2015, Figure 1). The entries are combined in blocks, which are linked by checksums (numbers to validate the integrity of data) to the previous block (Christidis and Devetsikiotis, 2016). This linkage as well as the decentralized and synchronized storage on all participating nodes of the blockchain network are supposed to make blockchains invulnerable to subsequent data manipulation and hacking. At the moment, blockchains find their widest application in storing transactions of crypto-currencies such as Bitcoin or Ethereum, but they can contain any sort of information, such as text, documents, images, or music.

Blockchains solve the problem that electronic transactions between two partners have to be mediated by a third authority, e.g. a bank, payment service, or seller (Tapscott and Tapscott, 2016). Instead, the transactions are verified by the blockchain network, which is self-organized and cannot be manipulated afterward. Therefore, blockchains are ideal in situations in which the trust of network participants into any mediating/controlling party is limited, i.e. that centralized administration of transactions is not trusted or technically not feasible. The latter case may arise when for example supply chains of sea food products cross the limits of multiple responsible controlling and surveillance authorities, e.g. by imports from Asia into the European Union or North American market. Or if any sort of regulating authority is totally absent, e.g. in remote, artisanal fisheries with unregistered fleets.

Smart contracts are self-executing scripts included into blockchains that automatize predefined operations, e.g. trading rules for crypto currencies or other assets (Christidis and Devetsikiotis, 2016). Smart contracts thereby extend the functionality of blockchains to allow for more complicated operations than just simple transfer of assets, e.g. conditional trades including “if-then” functions and other conditional rules. Further functionalities are constantly added to blockchain platforms and their technological development is far from finished. Current efforts are focusing on increasing the speed of blockchain transactions while reducing the energy requirements to maintain the network.

Data mining and big data

The global use of digital devices generating increasing amounts of data is growing fast. In 2015, presumably about 8 zettabytes of e-mails, blogs, social media posts, images, and videos were created

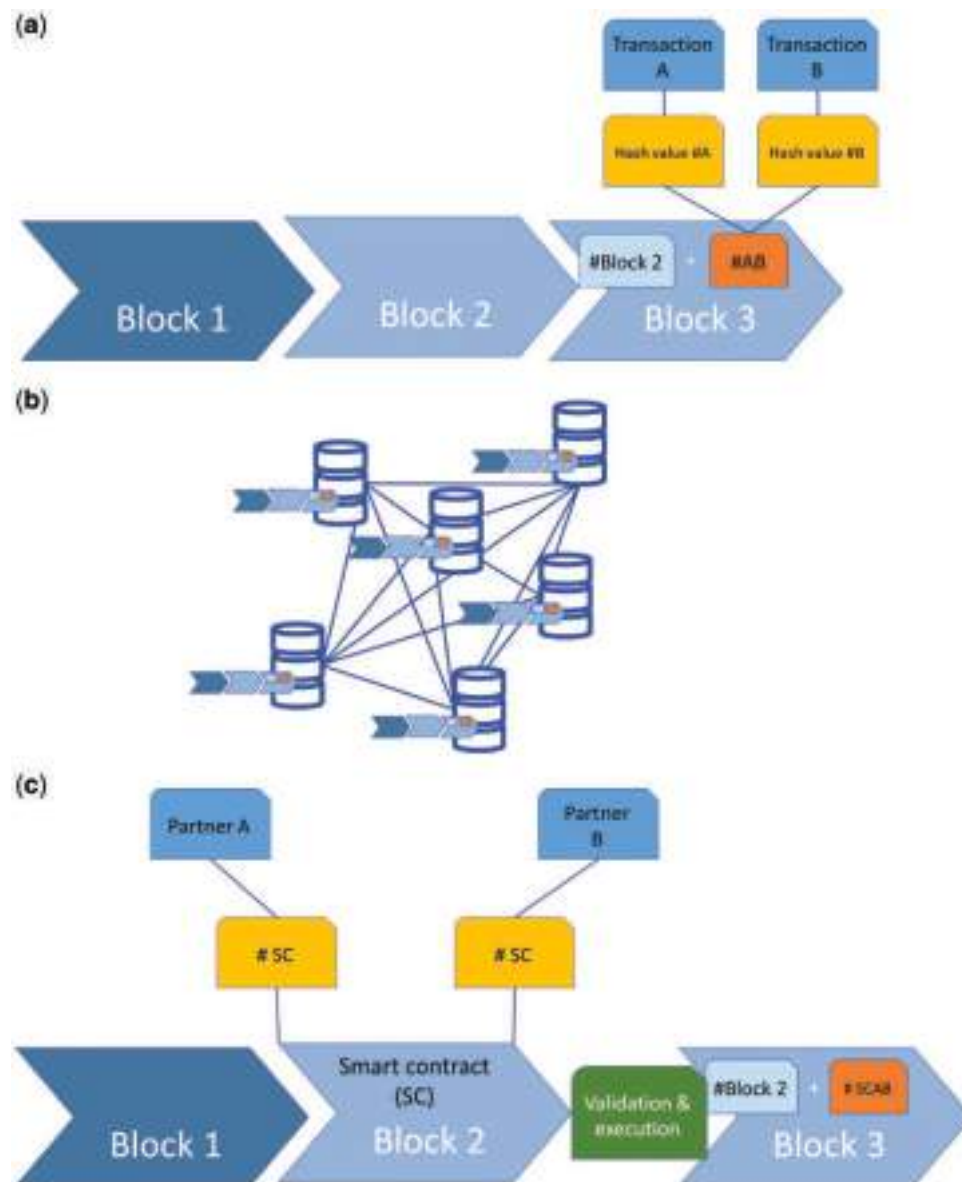


Figure 1. A simplified representation of blockchains. (a) Transactions are coded by hash values, which in turn are combined into blocks within the block chain. Single blocks are connected via the hash of the previous block and the hash values of new transactions. (b) The block chain is shared among the participants of the network, with each participant holding an identical copy of the blockchain. New blocks must be validated by a critical number of network participants to be added to the blockchain. When this happens, the blockchain version of all network members will be updated. (c) Smart contracts are implemented within the blockchain, i.e. are blocks that are referred to by contracting parties (here partners a and b). The smart contract validates the requested action and if approved, executes its content (e.g. a conditional transaction coded with the contract). The executed contract will be amended to the blockchain when approved by the network.

(Rajamaran, 2016). Accordingly, architectures and tools for storing, exchanging, handling, and analysing large volumes of data (“big data”) have been developed (Zakir *et al.*, 2015). The process of analysing big data is commonly referred to as data mining (Kantardzic, 2011) and includes methods to describe patterns within the data and to predict events from these data. Data mining itself is not a new branch of information technology, but it has been boosted with the demand for real-time analysis of large data volumes in web-based applications and digitized industries (Zakir *et al.*, 2015). Typical data mining tasks are the identification of outliers, classification, and clustering of data, regression

analysis as well as condensing data into summaries and overviews.

While many analysis in natural sciences traditionally include some form of data mining, in the context of this article data mining techniques are commonly associated with big data gathered for non-scientific purposes (Walker, 2014).

Big data are usually described by volume, commonly ranging in the terra- to petabyte domain, but big data can also be classified by variety and velocity (Russom, 2011; Rajamaran, 2016). Variety refers to the type of data available, e.g. structured, semi-structured, or unstructured, depending on the sources the data is

coming from and which data sources are combined. Classical sources for big-data analysis are data warehouses, in which several data bases of different format and content are combined. Velocity refers to the frequency of data streams, i.e. whether data are updated or added in real-time, near real-time, or in batches. Especially real-time data are most likely to have the strongest potential for innovative applications, but are also the most demanding with regards to storage, processing, and analysing.

Artificial intelligence

AI is a branch of computational sciences which deals with the ability of machines, i.e. computers to achieve goals by learning based on previous experience (Russell and Norvig, 2010). AI is therefore often used interchangeably with the term “machine learning”, even though AI is a more generic concept, of which machine learning is only one aspect. The ultimate goal of machine learning algorithms is to come to automatized decisions in non-determined situations resembling human cognitive abilities, such as recognizing objects in images or translating sentences from one language into another. AI finds application in speech and image recognition (Kantardzic, 2011; Ghahramani, 2015), crime prediction (Shapiro, 2017), or autonomous driving of vehicles (Urmson *et al.*, 2008). In the wake of big-data applications, machine learning algorithms have become very popular, as they are able to learn as they are trained on existing data (Hastie *et al.*, 2009; Tayal *et al.*, 2014; Ghahramani, 2015; Rajamaran, 2016; Shapiro, 2017). The major asset of machine learning algorithms is their ability to learn from the training data and to incorporate new data into the learning process as the data flow into the database.

Machine learning algorithms are often categorized into supervised and unsupervised learning algorithms (Hastie *et al.*, 2009). Supervised learning are based on some sort of model, in which data are separated into input and output features, i.e. into predictor and response variables (Kotsiantis, 2007). Some popular supervised machine learning algorithms are decision trees, support vector machines, Bayesian networks, boosted regression trees, random forests, and artificial neural networks. They can be applied for medical applications such as cancer prognosis and prediction (Kourou *et al.*, 2015) or the analysis of social media content (Ruths and Pfeffer, 2014). Unsupervised learning methods analyses the data structure without distinguishing between input and output variables, with many unsupervised algorithms working around clustering and discriminating data cases according to their features (Hastie *et al.*, 2009). A famous example of a non-supervised machine learning algorithm is the Google PageRank algorithm for web searches in the World Wide Web.

How can Blockchain & Co. improve the trust fisheries?

Blockchain

One of the few documented examples to implement blockchains in fisheries is the attempt to support the traceability of tuna around Fiji and other South Pacific islands (Visser and Hanich, 2017). To combat illegal, unreported, and unregulated (IUU) fishing, participating fisheries attach radiofrequency identification (RDFI) or quick-response (QR) code tags to each fish immediately after its catch. These tags are then registered automatically at various stations of the processing (i.e. supply) chain and each registration is fed into the blockchain. According to the example

of the South Pacific tuna fishery, a UK-based company called “Provenance” implements blockchains for agricultural, forestry, and fishing products including fair-trade and organic consumer labels (www.provenance.org).

These application exemplify several advantages of blockchains, i.e. the registration and processing of the product is automatized and the data in the blockchain is supposed to be tamper proof as it should not be modifiable by a single member of the supply chain (Pfreundt, 2018, but see section on problems on caveats for safety issues with blockchains). Furthermore, the blockchain does not have to be organized by a single controlling agency (non-governmental or governmental institution), but is self-organizing across jurisdictional borders and institutional responsibilities. And finally, the blockchain transactions are transparent, allowing all participants of the market, including the consumer, to track the origin of the fish. This does not mean that tracking of supply chains by blockchains are invulnerable to fraud and cheating. In the example of the South Pacific tuna fishery, the most crucial step within the supply chain is the correct labelling of individual fish. Mislabelling can still happen and with criminal energy, fishers, traders, ex-, and importers still may find ways to land and sell ill-declared fish. But the transactions registered within the blockchain will be unchangeable and hence cheating of a single party within the supply chain should become much more difficult.

Blockchains may help to monitor landed and traded fish along the supply chain much better than current, centralized databases, which are often restricted to national or regional jurisdictions. Data within blockchains could become a valuable source for enforcement agencies, fisher, traders, consumers, and scientists (Pauly and Zeller, 2003; Sumaila *et al.*, 2007; Mora *et al.*, 2009) to analyse catch and landing volumes as well as revenues of producers, processors, and traders. At the moment, these data have to be actively gathered by governmental agencies requiring significant funding and manpower (Stransky *et al.*, 2008; Dörner *et al.*, 2018). If landed fish and seafood would automatically register to a blockchain, landing volumes of a species could be counter checked with global retail volumes to identify and track discrepancies. Even if this sort of tracking would not work on a mandatory basis, as many participants in the fishing industry may lack the capacity or the will to participate in a fully electronical processing system, the participation could occur on a voluntary basis within consumer-labelling schemes, e.g. organic, sustainability or fair-trade labels (Gulbrandsen, 2009; Swan, 2015; Visser and Hanich, 2017). In such labelling schemes, landing shares as tonnes of biomass could be purchased by fishing companies to produce and trade these shares within the blockchain of the label. This would turn catch quotas into a blockchain asset similar to a crypto currency. Alternatively, governments (at least in developed countries) could combine the allocation of catch quotas to fisheries organizations with the obligation to apply a blockchain-based tracking system. Catches without a share in the blockchain could thus not be landed and sold, thereby becoming illegal.

When catch quotas are traded as blockchain assets, any quota share (e.g. 1.000 t of herring) would then be attributed with a volatile trade value, similar to stocks in a stock exchange. If the actual value of the fish within the quota was higher than the trade value of the quota, fishermen could decide to actually fish the quota out and the asset would be annulated from the market. If the trade value was higher than the price for the fish, fishermen could decide to keep or sell the quota as financial asset. This system would provide interesting options not only for fishing

companies, but also for non-governmental conservation organizations (NGO) or management authorities to implement buy-out schemes. The trade value of quota shares could rise if the annual catch quota is low, providing financial compensation to fishermen.

Finally, blockchains on landing trades may hold an advantage to fishermen themselves. If they would have access to all trades that have occurred at the port in their vicinity, they could choose the trader or port, which pays the best prices for their catch. Blockchains could also be used to trade catch quotas between fishing vessels, for example in fisheries with tradable catch quotas (Branch, 2009). Participating fishing vessels and management authorities would be informed on the ongoing trades in real time and also across national jurisdictions.

Data mining and artificial intelligence

Data mining and AI are tightly linked, as many AI algorithms require large datasets and some data mining techniques are in turn based on AI (Kotsiantis, 2007; Hastie *et al.*, 2009; Kantardzic, 2011). A prominent example of combined data mining and AI is predictive policing (Tayal *et al.*, 2014; Shapiro, 2017). Predictive policing uses georeferenced data (e.g. unemployment rate, population density, financial income of residents, etc.) to anticipate in which areas specific crimes such as burglary, mugging, or murder may occur at significantly high rates (Pearsall, 2013). Alternatively, some predictive policing software can estimate how likely previously convicted persons may commit another crime (Shapiro, 2017). Similar to predicting robberies and muggings in a city, it is possible to develop decision support tools based on AI and data mining for law enforcement agencies at sea. An example is a decision support tool for the US Coast Guard in the Gulf of Mexico to choose locations for patrolling against illegal fishing (Haskell *et al.*, 2014). This tool is based on a game-theoretic model which predicts the response of illegal fishermen to the patrolling scheme of the US Coast Guard in the Gulf of Mexico. While this example represents a situation with limited data availability on fishing activities, it is easy to envision even more powerful decision support tools in fully industrialized fishing fleets equipped satellite tracking devices and electronic logbooks. In these fisheries, reported catches and locations of fishing operations can be transmitted in near real time and analysed by data mining algorithms on land-based servers. If these algorithms detect significant deviations from the common pattern recorded for the according area, gear type, and season, patrols could be sent out to inspect these vessels at sea. This kind of system would be most efficient in a fisheries in which the majority of fishers comply with the management rules, as the data created by the “black sheep” would stand out from the majority of logbook records and thus should be easy to identify by data mining algorithms.

Data mining algorithms could be also used to improve the implementation of spatial real-time closure (RTC) in fisheries where high amounts of unwanted by-catch are observed. In 2009 the Scottish Government implemented a RTC scheme on areas of high catches of juvenile Atlantic cod *Gadus morhua* to reduce mortality and discarding (Holmes *et al.*, 2009; Needle and Catarino, 2011). In the RTC scheme, data from on-board observers and satellite-based ship tracking (VMS) are used to determine areas of high catches. However, data can only be analysed after the reporting of landings at harbour and thus the designation of closure sites is associated with a temporal delay. To improve the

implementation of RTC, catch data from e-logs could be transmitted to land-based servers, where they could be analysed by data mining algorithms even sooner.

A prominent application of AI algorithms is the field of image recognition. In fisheries, AI-based image recognition could improve electronic observing-at-sea programmes based on video monitoring. At the moment, the recognition of species and catch volumes from electronic monitoring is commonly done manually, requiring human observers to sight video footage and image stills (Kindt-Larsen *et al.*, 2011; van Helmond *et al.*, 2014). Consequently, only small proportions of the recorded data are actually analysed. The automatized recognition of species, length, and catch volume would allow to analyse more data at lower costs providing a more complete picture of the total removals of commercial fishing fleets. Some studies which have implemented AI-based image recognition to identify fish species in aquacultures (Hu *et al.*, 2012) and in catch fisheries (Storbeck and Daan, 2001), achieved classification rates of >90%.

Image recognition is also increasingly used to identify vessels at sea (Kanjir *et al.*, 2018). Optical remote sensing images can be combined with satellite or land-based geolocation techniques, such as vessel-monitoring system (VMS) or automatic identification system (AIS) to identify ship size and activity type. Though the satellite-based classification of vessels at sea is still faced with challenges by cloud cover, solar light input angle and overflight frequency, refined algorithms based on machine learning may help to overcome some of these shortcomings to detect unreported and illegally fishing vessels. Even if this data may not become available in real-time, similar to radar controls it may be used as evidence for fining and sanction schemes.

Apart from image recognition, AI may have many potential applications in fisheries and trading of seafood products. At the moment, implemented examples are scarce, but AI may find fruitful application in any situation in which large data volumes need to be analysed and categorised. Due to the versatility of AI and its potential applications, this article does not intend to provide a comprehensive overview on AI applications in fisheries, but rather intends to spark the creativity of software developers and decision makers in economic and management institutions.

A word on the “internet of things”

Internet of things (IoT) refers to the automatized communication of electronic devices (Christidis and Devetsikiotis, 2016). Its development is tightly linked to the equipment of devices with sensors and communication electronics such as Bluetooth or wireless network adapters. Recent IoT applications are smart home products that can regulate lighting, entertainment systems, heating and locking of doors and windows. There are also many applications in agriculture such as satellite-based harvesting and planting machines, automatized irrigation systems connecting soil sensors with pumps and water gates or radiofrequency identifiers (RFID) for tracking free-ranging cattle (Dlodlo and Kalezhi, 2015). Similarly, RFID have been attached to Tilapia reared in Chinese aquaculture to follow individual fish through the production and trading along the supply chain (Bo *et al.*, 2012). This is very similar to the blockchain tuna example described above, the only difference being that the Tilapia is tracked without a blockchain.

Without any doubt IoT applications will find many opportunities within fisheries productions and trades, but they are not in the focus of this review. While IoT and data applications are

Table 1. Examples of potential applications of blockchains, data mining, and AI in economics and management of fisheries.

Technology	Task	Strengths and opportunities	Weaknesses and threats
Blockchain	Tracking of the supply chain of fish and seafood products by consumers and/or retailers	Increase consumer trust and engagement in ecolabels and sustainability campaigns	Not fraud proof as mislabelling/mal-codification still may occur
Blockchain and smart contracts	Trading of quotas/shares among fishing companies and between management authorities and fishing companies	Transparent and self-organized trading of catch shares/quotas	May not comply with management specificities
Blockchain and data mining	Tracing and verifying reported landings by management authorities	Adds transparency to fisheries data Facilitated enforcement of fisheries regulations Allows to direct inspection efforts to critical cases	Difficult to implement and demanding on IT resources Algorithms may lead to wrong conclusions
	Sell at sea of catch by fishing companies to wholesalers factories, retailers	Improved trading opportunities for producers (fishers)	May incentivise the landing in countries with less strict management obligations
Data mining	Designation of RTC	Improved speed in closure designation	Requires near real-time transmission of catch information
Data mining and AI	Predictive control against IUU ^a , decision support for patrols	Improved allocation of inspection efforts, higher incentive for fishers to comply with laws and management rules	Requires near real-time transmission of catch information, which can still be flawed
AI	Catch recognition in on-board video footage	Improved processing of catch, better data collection for management authorities	Difficult to implement technically, may not be error or tamper-proof, high costs for companies operating fishing vessels
	Vessel identification of optical remote sensing images	Improved knowledge on vessel location and activity May have a deterrent effect on illegal fishing vessels	Spatial and temporal coverage may have significant gaps Error rate in vessel identification may be high

Note: This list is not intended to be comprehensive, but represents some potential applications together with associated strengths, weaknesses, opportunities, and threats.

^aIllegal, unreported, or unregulated.

tightly linked, the former is very much associated with hardware developments, i.e. the application of sensors in previously not measured systems. It can easily be imaged that IoT can play a big role in processing seafood on-board of fishing vessels, e.g. by improving automatized grading and gutting. Also a more direct communication between consumers and producers (fishermen) would be thinkable to enhance direct marketing of fresh caught fish, leading to better prices and short supply chains. However, these are just two of many possible IoT applications in fisheries, reviewing them all is beyond the scope of this article. Nonetheless it should be noted that several blockchains techniques are explicitly developed to accommodate requirements of IoT applications. A well-known example is the crypto currency IOTA, which the developers foresee as a native currency for financial transactions between autonomous devices (see www.iota.org).

Meta-view on potential applications of blockchain, data mining, and AI

Looking at Table 1, data mining and AI appear to be especially useful tools to monitor and control fishing vessels/companies by helping to gain knowledge on catches and to ensure compliance to management rules at sea. Thereby both techniques are acting mainly on the first link of the supply chain, the producer, i.e. the fishing vessel, whereas blockchains and smart contracts are useful to ensure transparency along the supply chain. It should be noted,

however, that the implementation of each technology is associated with caveats.

Problems and caveats

Innovations are usually two-sided medals and hence blockchain, data mining, and AI pose challenges to all participants of fishing enterprises including fishermen, trader, consumers, management authorities, and scientist. Fishermen may not be willing to instigate further mechanisms of control, whereas traders and management authorities may fear the extra costs and effort of installing and maintaining new infrastructures. And finally, consumers may need to engage into the blockchain by downloading apps and spending time to get informed on their product. Furthermore, neither blockchains, smart contracts, nor data mining algorithms are free of fraud, error and uncertainty. It is therefore naïve to assume that these technologies will entirely prevent illegal or unreported fishing. One of the most crucial steps of the fisheries supply chain will remain the haul of the catch on board of the fishing vessel and the subsequent designation of labels. Both processes will still remain in the hands of the vessel crew. Outside the fishing vessel, it will be difficult to verify whether all catches are labelled correctly, whether discarding has occurred or whether catches are landed unreported or illegally. Labelling also becomes easier to manipulate when fish and seafood products are not sold as whole, but are processed into different product categories such

as steaks, filets, loins and minced meat (Visser and Hanich, 2017).

The technical infrastructure to maintain blockchains in near real-time is challenging. Receiving and sending large data volumes at sea is only possible via satellite communication in many parts of the ocean. Installing satellite communication devices may be not feasible in small-scale fisheries or fisheries with limited financial resources. Transmission prices could still be too high and the available bandwidth still too narrow to support the transmission demands of electronic logbooks and synchronised blockchains.

Blockchains require intense and frequent communication between the nodes of the blockchain network. The synchronization of blockchains thus can be tardy (Christidis and Devetsikiotis, 2016) and may not work well in situations in which a large proportion of network nodes are faced with unstable network connections. Classical blockchains as used for Bitcoin are also faced with the challenge of scalability, i.e. a limited number of transactions that the network can process (Xu *et al.*, 2016). Thus vast expansion of network participants may pose challenges on the blockchain network.

Blockchains are also not fully tamper proof, as blockchain consensus could be reached by one party if it manages to hold the majority of the network nodes (Lin and Liao, 2017; Dey, 2018). While this may be an unlikely scenario in big blockchain networks, it may happen in smaller networks, when only a limited number of participants is involved.

Predictive policing is commonly criticized for loss of privacy and civil rights, i.e. introducing ethnical bias into the identification of crime sites or individuals (Shapiro, 2017). Accordingly, fishermen may find their privacy interests breached when being controlled by digital surveillance. Repeated protest against video monitoring at-sea and satellite-based VMSs in Europe indicate reluctance of fishermen to implement technologies which increase surveillance on their operations (Mangi *et al.*, 2015). Thus data technologies should find their fastest and most widely accepted implementation in situations, in which fishermen are incentivised to do so, e.g. if they gain trade benefits or are relieved from paper work.

Blockchains often require some sort of incentive for participation to maintain the network, which in cryptocurrencies is a mining reward. Obtaining this reward can be very energy consuming and costly, making an equitable participation of small and large stakeholders less likely. However, technical innovations of blockchain technologies are constantly produced and future applications may solve many of the aforementioned problems (Xu *et al.*, 2016; Christidis and Devetsikiotis, 2016). For example, processing energy requirements can be reduced when blockchain systems are based on proof-of-stake algorithm instead of the proof-of-work algorithm (Christidis and Devetsikiotis, 2016; Lin and Liao, 2017). Proof-of-stake allocates the computation of new blocks in a deterministic way, e.g. based on the amount of currency held by a participant of the network. Contrary, proof-of-work is a competing scheme in which the potential creators of new blocks (i.e. miners) are competing by finding the fastest solution to a complex puzzle, which requires significant amounts of computational power and thus energy.

Conclusions

This article can only sketch some potential applications of blockchains, big-data analysis and AI in fisheries. Currently

implemented examples and existing literature are too scarce to provide an in-depth review. Each technology will most likely evolve further, leading to unforeseen opportunities (and risks). Thus many of the described applications may never become realized, or their implementation may come with drawbacks which are not yet to be foreseen. But it is unlikely that the economics and management of fisheries will not be significantly affected by any of these technologies. Thus it is rather question of when and how enhanced data technologies will find entrance into the world's fisheries.

Blockchain, data mining, and AI will not stop IUU fishing, will not prevent overfishing and discarding. But they may help to make global streams of fish and seafood products with the associated flow of money becoming more visible and transparent. In fact, digital data technologies may work best in fisheries, which voluntarily intend to demonstrate their compliance to laws, management rules, and consumer demands or which are looking for a self-controlling mechanism to foster trust amongst competitors. Such systems may even evolve in areas, where governmental fisheries is currently weakly developed or totally absent, because fishermen may want to organize themselves to reduce conflicts and improve trade opportunities. Finally, in many situations, these technologies might allow governmental authorities to improve surveillance of industry compliance and consumers to place better informed decisions on which product they would like to purchase.

Acknowledgements

I thank Henrike and Phillip Rambo, who for the first time made me aware of the implications of crypto-currencies and blockchains. Two anonymous reviewers and the editors of *ICESJMS* provided valuable comments on the first version of the manuscript.

References

- Benoit, H. P., and Allard, J. 2009. Can the data from at-sea observer surveys be used to make general inferences about catch composition and discards? *Canadian Journal of Fisheries and Aquatic Sciences*, 66: 2025–2039.
- Bo, Y., Ping, S., and Huang, G. 2012. Development of traceability system of aquatic foods supply chain based on RFID and EPC internet of things. *Transactions of the Chinese Society of Agricultural Engineering*, 29: 172–183.
- Branch, T. A. 2009. How do individual transferable quotas affect marine ecosystems? *Fish and Fisheries*, 10: 39–57.
- Cath, C., Wachter, S., Mittelstadt, B., Taddeo, M., and Floridi, L. 2018. Artificial intelligence and the 'good society': the US, EU, and UK approach. *Science and Engineering Ethics*, 24: 505–528.
- Christidis, K., and Devetsikiotis, M. 2016. Blockchains and smart contracts for the internet of things. *IEEE Access*, 4: 2292–2303.
- Dey, S. 2018. A proof of work: securing majority-attack in blockchain using machine learning and algorithmic game theory. *International Journal of Wireless and Microwave Technologies*, 5: 1–9.
- Dlodlo, N., and Kalezhi, J. 2015. The internet of things in agriculture for sustainable rural development. *In Proceedings of the International Conference on Emerging Trends in Networks and Computer Communications*, Windhoek, Namibia, 17 May–20 May 2015. Institute of Electrical and Electronics Engineers (IEEE), Windhoek, Namibia.
- Dörner, H., Casey, J., Carvalho, N., Damalas, D., Graham, N., Guillen, J., Holmes, S. J. *et al.* 2018. Collection and dissemination

- of fisheries data in support of the EU Common Fisheries Policy. *Ethics in Science and Environmental Politics*, 18: 15–25.
- Edgar, G. J., Bates, A. E., Bird, T. J., Jones, A. H., Kininmonth, S., Stuart-Smith, R. D., and Webb, T. J. 2016. New approaches to marine conservation through the scaling up of ecological data. *Annual Review of Marine Science*, 8: 435–461.
- FAO. 2016. *The State of World Fisheries and Aquaculture 2016. Contributing to Food Security and Nutrition for All*. Rome. 200 pp.
- Ghahramani, Z. 2015. Probabilistic machine learning and artificial intelligence. *Nature*, 521: 452–459.
- Gulbrandsen, L. H. 2009. The emergence and effectiveness of the Marine Stewardship Council. *Marine Policy*, 33: 654–660.
- Hall, M. A. 1996. On bycatches. *Reviews in Fish Biology and Fisheries*, 6: 319–352.
- Haskell, W. B., Kar, D., Fang, F., and Tambe, M. 2014. Robust detection of fisheries with COMPASS. In *Proceedings of the Twenty-Sixth Annual Conference on Innovative Applications of Artificial Intelligence*, pp. 2978–2983. Association for the Advancement of Artificial Intelligence, Québec City, Canada.
- Hastie, T., Tibshirani, R., and Friedman, J. (2009) *The Elements of Statistical Learning*, Springer Series in Statistics, Springer, New York.
- Haug, S., Castro, R. P., Kwon, M., Filler, A., Kowatsch, T., and Schaub, M. P. 2015. Smartphone use and smartphone addiction among young people in Switzerland. *Journal of Behavioral Addiction*, 4: 299–307.
- Helyar, S. J., Lloyd, H. A., de Bruyn, M., Leake, J., Bennett, N., and Carvalho, G. R. 2014. Fish product mislabelling: failings of traceability in the production chain and implications for illegal, unreported and unregulated (IUU) fishing. *PLoS One*, 9: e98691.
- Herr, H., Fock, H. O., and Siebert, U. 2009. Spatio-temporal associations between harbour porpoise *Phocoena phocoena* and specific fisheries in the German Bight. *Biological Conservation*, 142: 2962–2972.
- Holmes, S. J., Campbell, N., and Aires, C. 2009. Using VMS and fishery data in a real time closure scheme as a contribution to reducing cod mortality and discards. *ICES Document CM 2009/M*: 13. p. 27.
- Hu, J., Li, D., Duan, Q., Han, Y., Chen, G., and Si, X. 2012. Fish species classification by color, texture and multi-class support vector machine using computer vision. *Computers and Electronics in Agriculture*, 88: 133–140.
- Jacquet, J. L., and Pauly, D. 2007. The rise of seafood awareness campaigns in an era of collapsing fisheries. *Marine Policy*, 31: 308–313.
- Kaiser, M. J., Clarke, K. R., Hinz, H., Austen, M. C. V., Somerfield, P. J., and Karakassis, I. 2006. Global analysis of response and recovery of benthic biota to fishing. *Marine Ecology Progress Series*, 311: 1–14.
- Kanjir, U., Greidanus, H., and Ostir, K. 2018. Vessel detection and classification from spaceborn optical images: a literature survey. *Remote Sensing of Environment*, 207: 1–26.
- Kantardzic, M. (2011) *Data Mining - Concepts, Models, Methods, and Algorithms*, 2nd edn. Wiley, Hoboken, NJ.
- Kindt-Larsen, L., Kirkegaard, E., and Dalskov, J. 2011. Fully documented fishery: a tool to support a catch quota management system. *ICES Journal of Marine Science*, 68: 1606–1610.
- Kotsiantis, S. B. 2007. Supervised machine learning: a review of classification techniques. *Informatica*, 31: 249–268.
- Kourou, K., Exarchos, T. P., Exarchos, K. P., Karamouzis, M. V., and Fotiadis, D. I. 2015. Machine learning applications in cancer prognosis and prediction. *Computational and Structural Biotechnology Journal*, 13: 8–17.
- Kwon, M., Lee, J.-Y., Won, W.-Y., Park, J.-W., Min, J.-A., Hahn, C., Gu, X. *et al.* 2013. Development and validation of a smartphone addiction scale (SAS). *PLoS One*, 8: e56936.
- Lewison, R., Crowder, L., Read, A., and Freeman, S. 2004. Understanding impacts of fisheries bycatch on marine megafauna. *Trends in Ecology & Evolution*, 19: 598–604.
- Li, S., Xu, L. D., and Zhao, S. 2015. The internet of things: a survey. *Information Systems Frontiers*, 17: 243–259.
- Lin, I.-C., and Liao, T.-C. 2017. A survey of blockchain security issues and challenges. *International Journal of Network Security*, 19: 653–659.
- Mangi, S. C., Dolder, P. J., Catchpole, T. L., Rodmell, D., and de Rozarieux, N. 2015. Approaches to fully documented fisheries: practical issues and stakeholder perceptions. *Fish and Fisheries*, 16: 426–452.
- Mora, C., Myers, R. A., Coll, M., Libralato, S., Pitcher, T. J., Sumaila, R. U., Zeller, D. *et al.* 2009. Management effectiveness of the world's marine fisheries. *PLoS Biology*, 7: e1000131.
- Mosler, H.-J. 1993. Self-dissemination of environmentally responsible behavior: the influence of trust in a common dilemma game. *Journal of Environmental Psychology*, 13: 111–123.
- Needle, C. L., and Catarino, R. 2011. Evaluating the effect of real-time closures on cod targeting. *ICES Journal of Marine Science*, 68: 1647–1655.
- Pauly, D., and Zeller, D. (2003) The global fisheries crisis as a rationale for improving the FAO's database of fisheries statistics. In *From Mexico to Brazil: Central Atlantic Fisheries Catch Trends and Ecosystem Models*, pp. 1–9. Fisheries Centre Research Reports. Ed. by D. Zeller, S. Booth, E. Mohammed, and D. Pauly. University of British Columbia, Vancouver, Canada.
- Pearsall, B. (2013) Predictive policing: the future of law enforcement? *National Institute of Justice Journal*, 266: 16–19.
- Pfreundt, U. 2018. How to harness blockchain technology for marine conservation. *PeerJ Preprints*, 6: e26496v2.
- Rajamaran, V. 2016. Big data analytics. *Resonance*, 21: 695–716.
- Russell, S. J., and Norvig, P. (2010) *Artificial Intelligence: A Modern Approach*. Pearson, New York, 37 pages.
- Russom, P. 2011. Big data analytics. TDWI Best Practice Report, 37.
- Ruths, D., and Pfeffer, J. 2014. Scoial media for large studies of behavior. *Science*, 346: 1063–1064.
- Samaha, M., and Hawi, N. S. 2016. Relationships among smartphone addiction, stress, academic performance, and satisfaction with life. *Computers in Human Behavior*, 57: 321–325.
- Schröder, U. 2008. Challenges in the traceability of seafood. *Journal für den Verbraucherschutz und Lebensmittelsicherheit*, 3: 45–48.
- Shapiro, A. 2017. Reform predictive policing. *Nature*, 541: 458–460.
- Storbeck, F., and Daan, B. 2001. Fish species recognition using computer vision and neural network. *Fisheries Research*, 51: 11–15.
- Stransky, C., Berkenhagen, J., Berth, U., Ebeling, M., Jiménez-Krause, J. D., Panten, K., and Schultz, N. 2008. National fisheries data collection programme: activities and outlook. *Informationen aus der Fischereiforschung*, 55: 5–14.
- Sumaila, U. R., Marsden, A. D., Watson, R., and Pauly, D. 2007. A global ex-vessel fish price database: construction and applications. *Journal of Bioeconomics*, 9: 39–51.
- Swan, M. (2015) *Blockchain: Blueprint for a New Economy*. O'Reilly Media, Sebastopol, CA.
- Tapscott, D., and Tapscott, A. (2016) *Blockchain Revolution: How the Technology behind Bitcoin Is Changing Money, Business, and the World*. Penguin, New York.
- Tayal, D. K., Jain, A., Arora, S., Agarwal, S., Gupta, T., and Tyagi, N. 2014. Crime detection and criminal identification in India using data mining techniques. *AI & Society*, 30: 117–127.
- Ulleweitt, J., Stransky, C., and Panten, K. 2010. Discards and discarding practices in German fisheries in the north Sea and Northeast Atlantic during 2002–2008. *Journal of Applied Ichthyology*, 26: 54–66.
- Urmson, C., Anhalt, J., Bagnell, D., Baker, C., Bittner, R., Clark, M. N., Dolan, J. *et al.* 2008. Autonomous driving in urban environments: boss and the urban challenge. *Journal of Field Robotics*, 25: 425–466.

- van Helmond, A. T. M., Chen, C., and Poos, J. J. 2014. How effective is electronic monitoring in mixed bottom-trawl fisheries? *ICES Journal of Marine Science*, 72: 1192–1200.
- Visser, C., and Hanich, Q. A. 2017. How blockchain is strengthening tuna traceability to combat illegal fishing. *The Conversation*, 4, 4 pages.
- Walker, S. J. 2014. Big Data: a revolution that will transform how we live, work, and think. *International Journal of Advertising*, 33: 181–183.
- Xu, X., Pautasso, C., Zhu, L., Gramoli, V., Ponomarev, A., and Chen, S. 2016. The blockchain as a software connector. *In Proceedings of the 13th Working IEEE/IFIP Conference on Software Architecture (WICSA) Venice, Italy, 5–8 April 2016*. Institute of Electrical and Electronics Engineers (IEEE), Venice, Italy.
- Zakir, J., Seymour, T., and Berg, K. 2015. Big data analytics. *Issues in Information Systems*, 16: 81–90.

Handling editor: Cigdem Beyan

Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Original Article

Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system

Ahmad Salman^{1*}, Shoaib Ahmad Siddiqui², Faisal Shafait¹, Ajmal Mian³, Mark R. Shortis⁴, Khawar Khurshid¹, Adrian Ulges⁵, and Ulrich Schwanecke⁵

¹School of Electrical Engineering and Computer Science, National University of Sciences and Technology (NUST), Sector H-12, Islamabad 4400, Pakistan

²German Research Center for Artificial Intelligence (DFKI), Trippstadter Strasse 122, Kaiserslautern D-67663, Germany

³School of Computer Science and Software Engineering, The University of Western Australia, 35 Stirling Highway, Crawley, WA 6009, Australia

⁴School of Science, RMIT University, GPO Box 2476, Melbourne, VIC 3001, Australia

⁵Faculty of Design—Computer Science—Media (DCSM), RheinMain University of Applied Sciences, Unter den Eichen 5, Wiesbaden D-65195, Germany

*Corresponding author: tel: +92 0 51 90852559; fax: +92 0 51 8317363; e-mail: ahmad.salman@seecs.edu.pk.

Salman, A., Siddiqui, S. A., Shafait, F., Mian, A., Shortis, M. R., Khurshid, K., Ulges, A., and Schwanecke, U. Automatic fish detection in underwater videos by a deep neural network-based hybrid motion learning system. – ICES Journal of Marine Science, 77: 1295–1307.

Received 13 November 2018; revised 23 January 2019; accepted 24 January 2019; advance access publication 27 February 2019.

It is interesting to develop effective fish sampling techniques using underwater videos and image processing to automatically estimate and consequently monitor the fish biomass and assemblage in water bodies. Such approaches should be robust against substantial variations in scenes due to poor luminosity, orientation of fish, seabed structures, movement of aquatic plants in the background and image diversity in the shape and texture among fish of different species. Keeping this challenge in mind, we propose a unified approach to detect freely moving fish in unconstrained underwater environments using a Region-Based Convolutional Neural Network, a state-of-the-art machine learning technique used to solve generic object detection and localization problems. To train the neural network, we employ a novel approach to utilize motion information of fish in videos via background subtraction and optical flow, and subsequently combine the outcomes with the raw image to generate fish-dependent candidate regions. We use two benchmark datasets extracted from a large Fish4Knowledge underwater video repository, Complex Scenes dataset and the LifeCLEF 2015 fish dataset to validate the effectiveness of our hybrid approach. We achieve a detection accuracy (F-Score) of 87.44% and 80.02% respectively on these datasets, which advocate the utilization of our approach for fish detection task.

Keywords: deep learning, fish assemblage, fish detection, fisheries management, neural networks, stock assessment, underwater video

Introduction

Monitoring the effect of preventive and recovery measures requires the estimation of fish biomass, and abundances by sampling their populations in water bodies like lakes, rivers and oceans on a regular basis (Jennings and Kaiser, 1998). This requires observation of the interaction of different fish species with changing environmental conditions. This is an essential process, especially in those regions of the world where certain species

of fish are either threatened or at the risk of extinction due to habitat loss and modification, industrial pollution, deforestation, climate change, and commercial overfishing (Tanzer *et al.*, 2015). There is a well-established and increasing interest in using non-destructive fish sampling techniques by marine biologists and conservationists (McLaren *et al.*, 2015). Underwater video-based fish detection approaches have been used to achieve non-destructive and repeated sampling for many years (Harvey and

Shortis, 1995; Shortis *et al.*, 2009). Manual processing of underwater videos is labour intensive, time consuming, expensive and prone to fatigue errors. In contrast, automatic processing of the underwater videos for fish species classification and biomass measurement is an attractive alternative. However, high variability in underwater environments due to changes in lighting conditions, clarity of water, and background confusion due to vibrant seabed structure pose great challenges towards automatic detection of fish. These factors result in a compromise on accuracy, which supports the continuing practice of less cost effective and cumbersome manual sampling and tagging of fish.

In general, automatic fish sampling involves the following three major tasks: (i) Fish detection, which discriminates fish from non-fish objects in underwater videos. Non-fish objects include coral reefs, aquatic plants, sea grass beds, sessile invertebrates such as sponges, gorgonians, ascidians, and general background. (ii) Fish species classification, which identifies the species of each detected fish from the predetermined pool of different species (Siddiqui *et al.*, 2017). (iii) Fish biomass measurement, using length to biomass regression methods (Froese, 2006). This article addresses the first task and the interested reader is referred to the literature for details of the following two steps in the overall process.

Various approaches have been followed for fish detection and consequently their assemblage estimation using image and video processing algorithms. Broadly speaking, these approaches can be divided into two categories based on the medium available for sampling, namely constrained and unconstrained sampling. In the former case, early attempts were made that involved detection of fish using information of their shape and colour (Strachan and Kell, 1995) or 3D modelling of fish to acquire features like height, width, or thickness (Storbeck and Daan, 2001). Harvey and Shortis (1995) presented an approach to acquire underwater images of fish under controlled conditions. This was achieved by making fish swim through a chamber with controlled illumination. Unconstrained underwater fish detection and classification does not assume any specific environmental conditions and, therefore, faces difficulty in achieving the required accuracy due to high variations in the aforementioned conditions. To address this problem, Spampinato *et al.* (2008) presented an image processing based method for fish detection and counting by capturing the texture pattern of fish in the natural underwater environment. They were able to achieve an average accuracy of about 84% on five underwater videos. In the past, several attempts have been made to solve the same problem in underwater videos using machine learning. Principal component analysis (Turk and Pentland, 1991), linear discriminant analysis (Mika *et al.*, 1999), and sparse representation-based detection (Hsiao *et al.*, 2014) presented some ways to capture fish-dependent features through mathematical modelling, which assumed independence of modelled fish with surrounding environments in videos. In other words, information like fish colour, texture, and shape was extracted with the prior assumption that foreground fish instance was easily distinguishable from the background. In reality, it is challenging to differentiate fish within underwater video/images due to camouflage with the background, poor visibility, and loss of contrast as a result of light attenuation through the water medium, low light, and water turbidity. In pursuit of suppressing the effects of environmental variability, Kernel Descriptors in Kernel density estimation (KDE) approach with colour information for background pixel modelling in images were used by

Sheikh and Shah (2005). In contrast, texture-dependent features computed via local binary patterns for background modelling was proposed in Yao and Odobez (2007).

Background modelling is a popular technique to segment moving foreground objects from the background in video sequences. An approach using motion-based fish detection in videos was presented by Hsiao *et al.* (2014). This method implements background subtraction by modelling background pixels in the video frames using Gaussian mixture models (GMMs). Although training the GMM, it is assumed that subsequent frames of video lack fish instances. Motion is detected in the video frames (apparently from fish) when a certain region of the frame does not fit into the trained background model. This approach produces fish detection results with an average success rate of 83.99% on several underwater videos collected near southern Taiwan. A similar scheme was proposed on covariance modelling of background and foreground (fish instances) in the video frames using colour and texture features of fish (Palazzo and Murabito, 2014). Using a dataset of four underwater videos with a high variation in luminosity, strong background movements, dynamic textures, and rich background, they were able to achieve an average detection accuracy of 78.01%. Presently, GMM- and KDE-based fish detection approaches are considered state-of-the-art (Spampinato *et al.*, 2014). We will compare the performance of various state-of-the-art techniques with our proposed approach in a later section.

All of the above-mentioned machine learning and feature extraction approaches fall into the category of shallow learning architectures (Bengio, 2009). These techniques are unable to accurately model the complexity of fish-dependent features in the presence of highly variable and diverse environments, and therefore these video or image-based fish detection techniques exhibit low performance in real-world scenarios (Siddiqui *et al.*, 2017). In the last decade, deep learning has been at the centre of attention for many researchers developing detection and classification algorithms in computer vision. Marked by their ability to extract and model highly nonlinear data, deep architectures have been utilized in numerous tasks related to computer vision, including facial recognition, speech processing, generic object detection, and classification in video and still images producing state-of-the-art results (Lin *et al.*, 2015; Ren *et al.*, 2017). In realizing deep architectures, multilayer deep neural networks are among the most successful schemes capable of extracting task-dependent features in the presence of variability in the images. Most commonly used variants of deep neural networks include deep convolutional neural networks (CNNs) which are parametric neural network models capable of extracting task-specific features and are widely used in computer vision problems like object recognition in images and facial recognition (LeCun *et al.*, 2015).

Deep learning is being used lately to solve fish-related tasks (Moniruzzaman *et al.*, 2017). An important work using CNN was proposed by Sung *et al.*, (2017) to detect fish in underwater imagery with 65.2% average accuracy on a dataset containing 93 images having fish instances. The system was trained on raw fish images to capture colour and texture information for localizing and detecting fish instances in the images. In a similar work, deep region-based CNN (R-CNN) were used for the abundance estimation of fish from 4909 underwater images recorded in the coast of Southeast Queensland, Australia. In this work, an accuracy of 82.4% was reported using the R-CNN system tuned for

locating and detecting fish instances in an image with a unified network framework.

Despite the high accuracy achieved by the deep learning based fish species classification, the task of vision-based automatic fish detection in unconstrained underwater videos is still under extensive investigation as most of the previous attempts reported results on relatively small datasets with a limited variety in the surrounding environment. Therefore, it is important to judge the robustness and effectiveness of any system in a large dataset with a high degree of environmental variation.

In this article, we address fish detection in highly diverse and unconstrained underwater environments and propose a hybrid system based on explicit motion-based feature extraction followed by a final detection phase using deep CNNs. In the first step, we use background subtraction by modelling still pixels of the video frames using GMMs. These models represent pixels related to a range of coral reefs, seabed features, and aquatic plants. Foreground objects are segmented from the background based on the motion in the scene that does not match the background model. To enhance the quality of the extracted features in each video frame, we concatenate the GMM candidate output blobs with the moving candidates generated by optical flow, a well-established approach used for motion detection in videos (Brox *et al.*, 2004). However, due to poor image quality, noise and background confusion, the detection remains far from perfect. To address this problem, we tune the parameters of GMM and optical flow systems to generate high recall by trying various values of the number of Gaussian distributions, initial variance, blob size and sensitivity in case of GMM, as well as pyramid size, number of pyramid layers, and window size in case of optical flow. The details of these parameters are given in Zivkovic and Heijden (2006) for GMM and in Beauchemin and Barron (1995) for optical flow. Specifically, in this step, all entities that exhibit even a slight movement are detected as fish. In the second step, we discriminate all the candidate regions in the video frames as fish and non-fish entities using a CNN architecture arranged in a hierarchical fashion to fine tune the detection system. Our CNN is trained using a supervised training style in which the GMM and optical flow blobs acts as the input while ground truth blobs (given in the training data) acts as the desired output. We worked on two different datasets; the Fish4Knowledge Complex Scenes Dataset, where the aim is fish detection with videos arranged into seven different categories based on the variation in the underwater environment; and the LifeCLEF 2015 (LCF-15) dataset, which is also designed for the detection of freely swimming fish in video sequences. These datasets contain marine scenes and species; unfortunately, there is no public domain benchmark datasets available containing underwater recordings in fresh water bodies.

The contribution of this work is to overcome the main challenge faced by the conventional motion detection and image classification approaches using deep learning. These deep learning modules are trained to select the relevant information from the data and minimize confusion which contributes to false alarms or missed detections. This approach improves the detection and classification accuracies especially in the data marked by high environmental variability like unconstrained underwater videos of fish. Our novelty lies in the proposed hybrid setup to mine the relevant motion information content by pooling the information generated by GMM and optical flow and refining the outcome by deep CNNs. Our approach is capable of detecting fish in the video in its stationary or moving state with region-based feature

localization. This equips our detection system with motion-influenced temporal information that is not available otherwise, in order to enhance detection performance in cases where fish is occluded or camouflaged in the background.

Material and methods

Dataset

We use two benchmark datasets in our study, both of which are specially designed to provide a resource for testing algorithms for detection and classification of fish in images and video sequences and have been used for benchmarking a number of approaches. The first dataset is used for the fish detection task and is a collection of 17 videos under different environmental conditions (http://f4k.dieei.unict.it/datasets/bkg_modeling/). The second dataset is taken from the LCF-15 fish task (<http://www.imageclef.org/lifeclef/2015/fish>). This dataset contains 93 underwater videos comprising 15 different fish species. Both datasets are derived from a very large fish database called Fish4Knowledge (Fisher *et al.*, 2016). With over 700 000 underwater videos in unconstrained conditions, the Fish4Knowledge dataset has been collected over a period of 5 years to monitor the marine ecosystem of coral reefs around Taiwan. This region is home to one of the largest fish biodiversity environments in the world with more than 3000 fish species.

The first dataset, dubbed FCS (Fish4Knowledge with Complex Scenes) hereinafter, comprises seven sets of selected videos recorded in typical underwater conditions addressing complex variability in the scenes. Thereby, the environmental variations provide a major challenge for fish identification and are categorized as follows:

- (1) **Blurred**, comprising three low contrast, blurred videos.
- (2) **Complex background**, composed of three videos with rich seabed structures that provide a high degree of background confusion.
- (3) **Crowded**, in which three videos with a high density of moving fish in each video frame imposes specific challenges for fish detection techniques, especially when it comes to high recall and precision in the presence of occluding objects.
- (4) **Dynamic background**, in which two videos are provided with rich textures of coral reef background and moving plants.
- (5) **Luminosity variation** composed of two videos involving sudden luminosity changes due to surface wave action. This phenomenon can induce false positives in detection due to moving light beams.
- (6) **Camouflage foreground**, two videos are chosen, addressing the challenge of detecting fish camouflaged in the presence of textured and colourful background.
- (7) **Hybrid**, in which two videos are selected to show a combination of all the above-mentioned conditions of variability.

Table 1 summarizes the technical information regarding both datasets used in this article. For the FCS dataset, complexity is specifically depicted for all seven environmental conditions. The LCF-15 dataset is used to detect fish instances in the video i.e. to count all the fish in the video regardless of their species. Of the 93 videos given in LCF-15, 20 are used for training the computer

Table 1. Information about LCF-15 and FCS fish datasets.

Dataset	No. of videos	Format	Resolution	Frames/sec	No. of labelled fish instances
LCF-15	93	FLV	640 × 480, 320 × 240	24	42 493
FCS	17	FLV	640 × 480, 320 × 240	24.5	1 328



Figure 1. Sample images to illustrate the high variation in underwater environment. The first two rows depict seven categories of the FCS dataset from left to right top to bottom being *Blurred*, *Complex background*, *Dynamic background*, *Crowded*, *Luminosity variation*, *Camouflage foreground*, and *Hybrid*. The last row shows an excerpt from different videos of the LCF-15 dataset.

vision or machine learning modules, while the remaining 73 videos are set aside for testing/validating the developed algorithms. In total, there are 9000 annotated fish instances available in the LCF-15 training set, and 13 493 annotated instances for the test videos. All these videos are manually annotated by experts. Apart from videos, there are 20 000 labelled still images in LCF-15, where each image comprises of a single fish. These images can also be used to supplement the training set if required. Thus, in total there are 42 493 labelled fish instances in videos and still images in the LCF-15 dataset. The FCS dataset is also designed and used for the fish detection task. Therefore, ground truth is available for all moving fish, frame by frame in each video. There are a total of 1328 fish annotations available for the FCS dataset. [Figure 1](#) shows some video frames extracted from FCS and LCF-15 datasets exhibiting the variation in the surrounding environment, fish patterns, shape, size, and image quality.

Proposed algorithm

To perform fish detection, we propose a hybrid system based on the initial motion-based feature extraction from videos using GMM and optical flow candidates. These feature images are

combined with raw greyscale images and fed to the CNN system to mark final detected fish. Therefore, our proposed hybrid fish detection system is made up of three components i.e. GMM, optical flow and a CNN.

Gaussian mixture modelling

In machine learning, GMM is an unsupervised generative modelling technique to learn first and second order statistical estimates of input data features ([Stauffer and Grimson, 1999](#); [Zivkovic and Heijden, 2006](#)). This approach and its variants are frequently used in computer vision and speech processing tasks. GMM represents a probability density function $P(x_t)$ at time t of data x as a weighted sum of multiple individual normal distributions $\eta(x_i)$ for pixel i . Thereby, each density is characterized by the mean and covariance of the represented data. Using a combination of individual Gaussian densities, any probability distribution can be estimated with arbitrary precision ([Reynolds and Rose, 1995](#)). In our case, each pixel value with a fixed location in the video frame acts as a feature. Multiple such values from successive frames are combined to form a feature vector. As elaborated in [Figure 2](#), we end up with a total number of feature vectors that

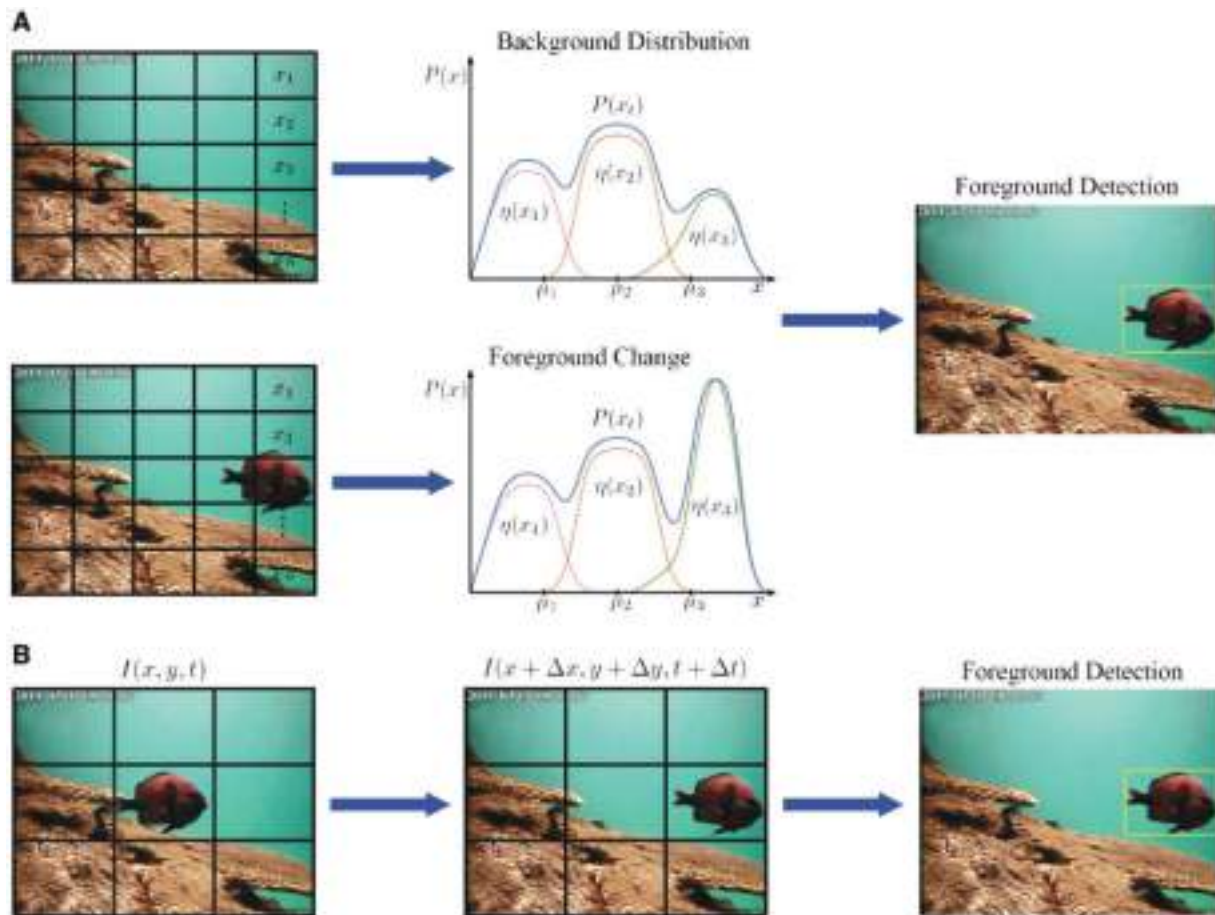


Figure 2. (A) Illustration of background subtraction and foreground segmentation using GMM which detects any change in the foreground by matching it with the background model. (B) Motion detection in an optical flow setup to estimate the direction of moving objects in two dimensions (x, y) for consecutive frames in time t of a video sequence.

equals the total number of pixels in a video frame. The GMM requires a certain amount of training data to effectively estimate the mean and covariance of an object class. For fish detection in videos, there are two classes i.e. background and foreground. Ideally, the background in underwater videos should cover everything in the frame but moving fish. For example, seabed structure, coral reef, aquatic plants, and wave action causing variation in light intensity are categorized as background. Freely moving fish, on the other hand, constitute as foreground. The GMM is used to learn the background features in a statistical model using mean and covariance values of pixel data and separate them from the foreground pixel distribution. In other words, any random and sudden change in the pixel value of a frame causes a mismatch with the background model of that pixel and hence, a motion is assumed to be detected. The statistical pattern of foreground (fish in our case) movement is usually different from the pattern of fixed objects like seabed structures, coral reefs and also objects with confined movement like to and fro motion of plants and refracted light rays from the surface. The outputs of the GMM are the candidate blobs marked by bounding boxes localizing the moving objects in a frame (see Figure 2).

The video frames that are used to train the GMM should not include any fish instance but only the background. However, it is

very difficult to capture such videos in a natural environment as fish can appear in any number of frames. When a GMM is trained on videos that do not have pure background but also some fish, the fish will also be modelled as background resulting in misdetections in the upcoming test frames.

Optical flow

To compensate for this shortcoming of GMM, we additionally extracted optical flow features which are purely generated by motion occurring in the underwater videos (see Figure 2). Optical flow is a 2D motion vector in the video footage caused by the 3D motion of the displayed objects (Warren and Strelow, 1985). There are various methods to estimate optical flow. We opted for a simple yet effective method where motion is detected between two successive video images taken at times t and $t + \Delta t$ at every position using Taylor series approximation with partial derivatives with respect to spatial and temporal coordinates (Beauchemin and Barron, 1995). A region of interest (ROI) in a video frame at time t and coordinates x, y can be represented in terms of intensity as $I(x, y, t)$. After any motion in the next frame, the intensity becomes $I(x + \Delta x, y + \Delta y, t + \Delta t)$ where the notation Δ represents the change in coordinates and time. Based

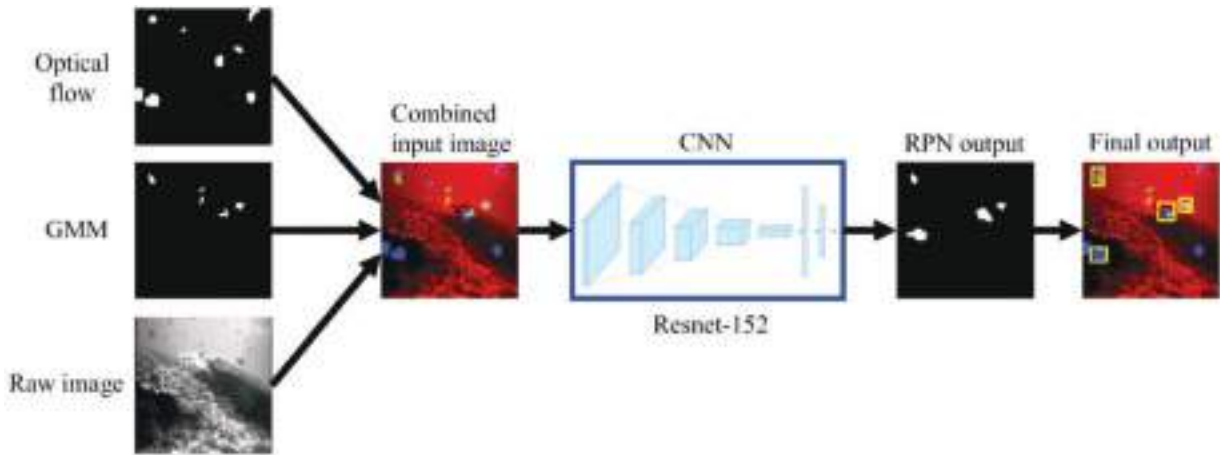


Figure 3. The proposed hybrid system, where ResNet-152 CNN is trained on images that are created by combining the motion-influenced outputs of GMM and optical flow algorithms with raw greyscale video images. This is analogous to three-channel RGB image.

on the motion constraint, optical flow can be determined as described in, for example, [Beauchemin and Barron \(1995\)](#).

Optical flow depends on the analysis of the consecutive frames to estimate the difference in the intensity vector of a pixel at a particular location. However, such an analysis is also prone to false motion detection apart from fish when applied to a dynamic background with moving aquatic plants and abrupt luminosity variation due to disturbance at the water surface. The parameters of the GMM and optical flow algorithm are chosen such that even the smallest movements are detected. In other words, the sensitivity of the algorithms is maximized, producing a high rate of false alarm in addition to detecting fish instances leading to high recall rates. In the next step, the precision of the system is further increased by fine-tuning and refining regions in the frames to localize moving fish. This requires a robust detector to categorize fish motion in complex and variable environments. We propose the use of a R-CNNs (hereinafter referred to as R-CNN) trained on images, created by combining candidate regions generated by the GMM and optical flow together with the original greyscale images in a supervised learning setup.

Region-based convolutional neural network

A deep CNN is a nonlinear parametric neural network model capable of extracting and learning complex yet abstract features of the input data. Variations in the lighting condition, size, shape, and orientation of the fish, poor image quality and significant noise are the factors that introduce nonlinearity into the data ([Bengio, 2009](#)). Since all of these challenges are encountered in the videos recorded in an unconstrained underwater scenario, it is difficult for conventional machine learning algorithms to model data features in the presence of such nonlinearity. However deep neural architectures, especially CNNs, learn to extract invariant and unique features of the objects of interest in data when properly trained with a sufficient amount of labelled data ([LeCun et al., 2004](#); [Simonyan and Zisserman, 2014](#)). The deep architecture exemplified by the R-CNN employed in our study is a hierarchical parametric model composed of two modules. The first module is a generic deep CNN trained for generic object recognition on a very large dataset called ImageNet ([Deng et al., 2009](#)). Smaller than the first module CNN, the second

module is another CNN, which acts as the object detector and called region proposal network (RPN) ([Ren et al., 2017](#)). It selects candidate regions in the feature space of the input image in which a motion is likely to have occurred.

The entire system is used for detecting moving objects as depicted in [Figure 3](#). The first module utilizes the concept of transfer learning ([Siddiqui et al., 2017](#)). It learns characteristic feature representation of the object of interest in the input image in order to recognize and classify the objects in the imagery. In transfer learning, a CNN pretrained on totally different, yet relevant dataset, is utilized as a generic feature extractor for the dataset of interest. In our case, the CNN was trained on the vast ImageNet dataset that contains 1.2 million images of a very large and diverse number of objects. This dataset is not related or designed for fish species recognition or fish detection in underwater videos. However, it provides a high degree of variability to detect generic objects with different backgrounds in input images based on their texture, size and shape features. Once the network is trained, it can be applied to a different dataset, in our case on underwater video imagery of fish, as a feature extractor. Transfer learning is suitable for the applications where a large amount of training data is not available to train the deep CNNs ([Siddiqui et al., 2017](#)). This is exactly the problem in the current underwater datasets. Training on such relatively small datasets (see [Table 1](#)) overfits a deep CNN to generate better performance on training dataset and fails on previously unseen test datasets. In other words, the training dataset is so small that the CNN is able to memorize it and produce good results only on the training dataset. We utilize a deep CNN known as ResNet-152 as the pre-trained model ([He et al., 2016](#)). The parameters of this network are further refined by including examples of our fish dataset video imagery in training. This network is composed of an input layer, various hidden layers and an output layer to process an input image to obtain its output feature representation ([LeCun et al., 2004](#)). Starting from an input layer that represents the pixels of an image, the hidden layers are interconnected by a set of weights that are tuneable as a result of the training procedure. Thereby, each hidden layer represents a higher-level form of feature representation. There are several types of hidden layers used in our network, e.g. a convolution layer that performs the mathematical operation of convolution between image pixels (values of the

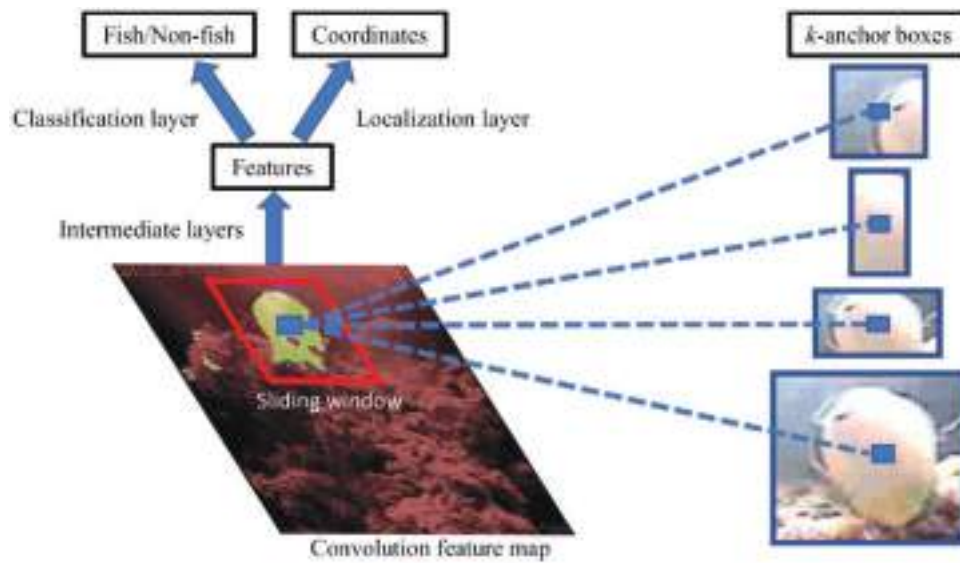


Figure 4. Illustration of the functionality of a RPN to detect and localize fish. The proposal with the best fit to the fish instance is selected out of k choices.

input layer) or feature values (values of the hidden layers) with the weight vectors. Convolution is generally used in image processing for noise reduction or detecting features such as edges or other patterns (LeCun *et al.*, 1989). In a CNN, convolution is followed by a nonlinear activation layer to induce nonlinearity in the feature vectors. There are several types of nonlinear functions, e.g. ReLUs (rectified linear units), Sigmoid and Hyperbolic Tangent (LeCun *et al.*, 2004; Simonyan and Zisserman, 2014; He *et al.*, 2016). The choice of the nonlinear function depends on the data distribution and nonlinearity of the input data. Due to the saturating regions of the Tangent Hyperbolic and Sigmoid function, the ReLU function is the defacto-standard in the latest state-of-the-art models. Max pooling and average pooling layers sift out the most prominent values from the output of nonlinearity inducing layers based on maxima or an averaging operations to reduce the dimension of feature vectors and retain useful information while discarding the redundancy. The final layer is the output layer which usually is a classification layer with output nodes equal to the number of desired classes for a given dataset. Each output node produces a score or probability for the associated class. The predicted label is then matched with the ground truth label to calculate accuracy.

ResNet-152 is a modular deep CNN with various hidden layers. The architecture is designed to process images of size 224×224 given the fact that this resolution is enough to extract useful features within reasonable computational time. Thus, after applying five pooling layers, the feature map size shrinks to 7×7 which can be processed by fully-connected layer of 1000 label prediction nodes, since ResNet-152 was designed to train on a subset of ImageNet dataset with 1000 classes.

The complete architecture details of ResNet-152 can be found in He *et al.* (2016). The arrangement of the above-mentioned layers in this architecture is experimentally determined to yield greater success on visual features from the large-scale ImageNet dataset. Using this network as a pretrained model on our FCS and LCF15 fish datasets, an informative visual representation of fish objects and their motion can be extracted. After applying the

pretrained ResNet-152 network on the input which is a concatenation of the raw greyscale video frames and the motion candidates generated by GMM and optical flow, we get the output features. This three-input combination is alternative to the standard three-channel RGB image. The output features extracted by applying ResNet-152 are fed into RPN to generate candidate regions where fish might be present. This is achieved by sliding a small window of size $3 \times 33 \times 3$ on each of the feature maps to produce k proposals, called anchor boxes, of different aspect ratio and scale. We use three different scales (128×128 , 256×256 , 512×512) each with 3 different aspect ratios (2:1, 1:1, and 1:2) to make $k = 9$ proposals. The aim of using different proposals is to capture fish of different sizes that may appear in an image. These proposals are then classified with a binary classification layer of the RPN to detect the ROI. Another sibling layer of RPN outputs coordinate encodings for each classified proposal. This operation is depicted in Figure 4. The ROIs proposed by the RPN are pooled using an ROI pooling layer and passed onto the final classification head which refines and classifies the proposed ROI into the actual number of classes present at hand, namely fish and non-fish. The complete network is trained in an end-to-end fashion using the features generated by ResNet-152 model as the input and the corresponding ground truths provided by the dataset. While training, we employ an error backpropagation algorithm (Hinton *et al.*, 2006).

As mentioned earlier, the parameters of the GMM-based motion detection algorithm are chosen such that it detects even a very small motion by either fish or non-fish objects producing high false alarm or recall rates. The R-CNN architecture, which is a combination of the ResNet-152 based feature extraction and RPN followed by a final classification layer for localizing moving objects, refines the output of the GMM and optical flow motion candidates. Therefore, the information of motion coming from GMM and optical flow is fed into R-CNN to finally detect and localize objects. Apart from motion candidates generated by GMM and optical flow, the use of greyscale raw images in combination with motion candidates as input to the ResNet-152 CNN helps in

Table 2. Performance analysis of individual components of our proposed hybrid framework in comparison to their joint accuracy.

Dataset	Optical flow		R-CNN	Our hybrid system
	GMM	flow		
FCS				
Blurred	77.80	45.94	85.62	86.76
Complex background	75.94	49.77	52.74	89.54
Crowded	74.41	67.48	53.23	84.27
Dynamic background	64.30	44.62	62.06	90.36
Luminosity change	59.07	58.67	70.17	81.44
Camouflage foreground object	70.03	67.00	66.25	89.97
Hybrid videos	75.50	59.44	64.90	91.50
Average	71.01	56.13	64.99	87.44
LCF-15	76.21	52.73	77.30	80.02

F-scores (in percentage) for three different methods i.e. GMM (Stauffer and Grimson, 1999), Optical flow (Warren and Strelow, 1985), and R-CNN (Ren et al., 2017) on FCS and datasets for seven categories of video complexity. Highest scores are highlighted in bold

preserving the textural information of fish appearing in the video frame, which increases the capability of the network to induce separability between fish and non-fish objects. The reason of using greyscale image instead of RGB to fine-tune the R-CNN is the observation that colour information in the employed datasets are not distinct enough to enhance the accuracy of detection as the background is also vibrant in colours. Moreover, doing so increases the computational overhead.

In this work, we utilized computer systems equipped with Intel Core-i7 processors and Nvidia Titan X graphical processing units (GPUs). The proposed system is trained and tested using TensorFlow deep learning library (<https://www.tensorflow.org>) with *Tf-faster-rcnn* version while GMM and optical flow source codes were taken from publicly available authors' repositories (<https://github.com/andrewsobral/bgslibrary>).

Fish detection system utility

Our software system is available for deployment and ready to be used by marine scientists for automatic fish detection in any dataset. As described in *Region-based convolutional neural network* Section, the deep network, which is the backbone of our algorithm, is pretrained on a large and generic object image repository called ImageNet and acts as a generic feature extractor. However, for using a pretrained network in such a transfer learning approach, the system must be fine-tuned to the actual datasets in hand; therefore, a complete end-to-end re-training on a new dataset is not required. In our case, we utilized FCS and LCF-15 datasets by updating the weights of the top fully connected layers of ResNet-152 of R-CNN, while keeping the lower layers intact. Furthermore, the GMM and optical flow algorithms can be used as is since they only require the available dataset to generate output. The source code for our proposed hybrid system is available for download from the following repository: <https://github.com/ahsan856jalal/Fish-Abundance>. Scientists can use this code off-the-shelf for fish/object detection in any dataset, video recordings or even still imagery.

Results

The underwater video background is modelled by GMM using training data from the initial few frames of the video while the remainder of the video is treated as the test dataset. Since each

video in our datasets has a different background, we need to keep the first N frames of each video for background modelling. We take $N = 50$ in our experiments as this value was chosen on a trial basis to get optimum GMM performance on our datasets. Smaller values of N produces an inferior performance, while increasing beyond this value does not bring any improvement and increases GMM training time. Optical flow does not require any training data but simply uses adjacent frames to calculate a motion representation. The R-CNN, on the other hand, requires more data to tune the weight parameters for refined motion detection. The raw video images and the motion candidates generated by the GMM and optical flow are fed to the R-CNN for training. One video from each of the seven categories of FCS dataset is set aside for training the GMM and R-CNN. On top of that, GMM also requires the first 50 frames of each video to make a background model and to generate a blob of moving objects in the test frames. The LCF-15 dataset, on the other hand, is already segmented into training and test sets, 20 videos out of a total of 93 are used in training and the remainder is used for testing. Once again, GMM models for all 93 videos are created using the N initial frames. Table 2 lists the performance measure for the fish detection task as an F-measure (Palazzo and Murabito, 2014) for our proposed hybrid system and its independent constituents of GMM, optical flow and standalone R-CNN, which are trained on raw RGB images from videos.

The F-score is calculated as,

$$F = \frac{2 \times \text{Recall} \times \text{Precision}}{\text{Recall} + \text{Precision}}$$

where

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

and

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

These scores are computed based on overlap between the areas of bounding boxes related to ground truths and detected fish. An average detection accuracy of 87.44% was achieved by the proposed hybrid system for the FCS dataset for all seven categories of environmental variation. In comparison, the GMM alone yielded an average accuracy of 71.01% exceeding the optical flow and standalone R-CNN with significant margins. We also performed similar experimentation on LCF-15 test dataset of 73 underwater videos. There, our proposed hybrid system outperforms all the other systems, yielding an accuracy of 80.02% as compared with 76.21, 52.73, and 77.30% by the GMM, optical flow and standalone R-CNN, respectively.

The parameters of the GMM were carefully chosen to produce best possible results by altering the variance for model fitting and the number of frames for training the model on each video. A fewer number of training frames per video results in degraded performance. However, increasing the number of training frames beyond 50 did not improve the overall performance significantly. Similarly, for our proposed hybrid system and also for the standalone R-CNN trained on the raw RGB images, various state-of-the-art pretrained CNNs were tried that include Inception-V4

(Szegedy *et al.*, 2016) and DenseNet (Huang *et al.*, 2017). All of these networks are pretrained on the ImageNet dataset with the same experimental settings. Moreover, different numbers of convolution layers for the RPN network were also evaluated and the choice of sliding window size of $2 \times 22 \times 2$, 4×4 , and 5×5 was tested, with the performance maximized at $3 \times 33 \times 3$. The performance started to deteriorate slightly beyond the $3 \times 33 \times 3$ window size probably due to more overlap between intrinsic size of fish covering the frame of videos in our datasets. The results generated by Inception-V4 and ResNet-152 were comparable without any significant difference but the latter utilizes less processing power in training and testing compared with the former. Our implementation of optical flow on the other hand is a non-trainable processing approach for motion detection and therefore, does not have any trainable parameters. It is worth mentioning here that the GMM chosen for our proposed hybrid system differs with the one listed in Table 2 as its parameters were tuned to produce higher recall rates at the cost of decreased precision to cover maximum possible pixel motion in the video by both fish and non-fish objects. The CNN and RPN subsystems then learn to select the relevant motion candidate through refining the results generated by the GMM and optical flow. Figure 5 shows the performance outcome on a sample video for GMM, optical flow, R-CNN, and the proposed hybrid system for both the FCS and the LCF-15 datasets. It is evident that the optical flow algorithm generates more false alarms and is sensitive to even very slight motion, which can be attributed to disturbances in the scene or luminosity changes. On the other hand, the GMM and stand-alone R-CNN, which is only trained on raw RGB images, also exhibits false alarms and/or missed detection. However, they both yield better scores as compared with the optical flow due to effective background modelling and end-to-end supervised training; capabilities which optical flow lacks and are necessary to reduce the irrelevant motion created by non-fish entities. Our proposed hybrid system, on the other hand is successful in achieving the best performance (see Table 2).

To validate the effectiveness of our system, in Table 3 we have drawn a comparison with various published benchmark approaches which are frequently used for motion-based object detection in either still or video imagery. The comparison is made on the FCS dataset for which we can directly tabulate published scores by these techniques with the same experimental settings as ours. It is evident that our proposed hybrid system outperforms all others in most environmental conditions and the overall average F-scores. In another set of experimentation not reported here, we changed the train-test split in the FCS and LCF-15 datasets to calculate the detection scores but observed no significant change. This demonstrates a good generalization capability of our system.

Discussion

In this study, we have proposed a R-CNN to detect fish using enhanced features sensitive to natural fish motion in underwater videos in addition to features also representing distinguishable shape and textural information specific to fish in a supervised training hierarchy. The motivation behind using such a deep neural network is to model complex and highly nonlinear attributes in underwater imagery of fish. These attributes are not modelled effectively by conventional machine learning algorithms and image processing techniques (Hinton and Salakhutdinov, 2006; Larochelle *et al.*, 2009). This hybrid approach has resulted in a

detection accuracy at reasonable level for use of this technique in fish detection from recorded videos.

The most important gain of this research is high detection accuracy of freely swimming fish. With our proposed hybrid system that incorporates motion sensitive features, taken as input to the R-CNN, we are able to achieve 87.44% detection accuracy on the FCS dataset. This performance exceeds the best reported results on this dataset by a significant margin. The second best average accuracy of 81.80% for all seven categories of variability has been produced using KDE to model background and foreground objects by capturing texture information in very low contrast regions of the video frames (Spampinato *et al.*, 2014). An interesting observation can be drawn from Table 3 for video classes *Dynamic background*, *Camouflage foreground object* and *Hybrid videos* that the performance gap between our proposed hybrid system and rest of the techniques is significantly wide. *Dynamic background* videos exhibit disturbance in water surface and movement of aquatic plants which causes confusion with motion of fish. Therefore, KDE, ML-BKG, and TKDE algorithms, which are based on estimating foreground data distribution by modelling background data, fails in separating motion of fish and non-fish objects. EIGEN and VIBE algorithms also produced poor performance due to similar reasons. Here, our proposed hybrid system utilizes the fish-dependent features captured through the R-CNN component using greyscale images in accurate detection of fish. On the other hand, fish in *Camouflage foreground object* videos are extremely hard to segregate from the background. Therefore, all the algorithms once again fail to yield better results due to inability in creating difference between foreground and background models. Here, our approach makes use of the motion information from GMM and optical flow to maximize its fish detection potential as shape, texture and colour of fish in this case resemble the background and are difficult to detect by the R-CNN component. Similarly, *Hybrid videos* combine all the challenges of other six classes and our proposed hybrid system is more effective than all other approaches. To further endorse the effectiveness of our approach, we employed a larger dataset by including LCF-15 with 93 videos. Our solution acquired an average accuracy of 80.02%. Table 2 lists the comparative performance of our proposed hybrid system with three other techniques, namely GMM, optical flow and R-CNN, which are the components of our overall system. The GMM outperforms optical flow and standalone R-CNN, trained on raw images, with a significant margin, for the FCS dataset. On the LCF-15 dataset, the GMM produces better results than optical flow and is comparable with the R-CNN. This signifies effective learning of the background model by the GMM on every new video sequence. The model covers all background variations exhibited by non-fish objects for a static underwater camera configuration, which assists in detecting even subtle movements through non-uniform change in pixel intensities that does not match with the distribution of background pixels.

We observe that the training of the GMM background model balances the rate of false alarm and misdetection, which produces a better F-score. The GMM and its variants are considered to give excellent performance in general for motion-based object detection tasks (Yong, 2013; Spampinato *et al.*, 2014). Optical flow, on the other hand, lagged behind all other methods in terms of performance on both datasets. The reason behind this behaviour can be attributed to the non-trainable structure of this algorithm, as the system cannot adapt to the dynamic environment in the videos. There is no learning involved to discriminate background



Figure 5. Example of fish detection outcomes by various algorithms. Left to right, ground truth, optical flow, GMM, stand-alone R-CNN, and proposed hybrid system on all seven categories of FCS dataset category (the first seven rows) and one video of LCF-15 dataset (the last row).

and foreground modelling, like that in the GMM and neural networks. Optical flow involves a direct comparison between adjacent frames of video and any slight disturbance in the pixel intensity, either due to fish or non-fish objects generating luminosity variation, translates into a valid motion. This gives rise to numerous false alarms, which results in a very high recall

but consequently a low precision that ends up in producing low F-score. Since the datasets we have chosen involve high environmental variation, especially in the FCS dataset, optical flow fails to perform well as opposed to the other algorithms. On comparative grounds, both GMM and optical flow lags our proposed hybrid system for the FCS and LCF-15 datasets. Another

Table 3. *F*-scores (in percentage) for different methods on FCS datasets for fish detection, as given in Spampinato *et al.* (2014).

Video class	KDE	ML-BKG	EIGEN	VIBE	TKDE	Our hybrid system
Blurred	92.56	70.26	81.71	85.13	93.25	86.76
Complex background	87.53	83.67	74.78	74.17	81.79	89.54
Crowded	82.46	79.81	73.87	84.64	84.19	84.27
Dynamic background	59.13	77.51	71.48	67.01	75.59	90.36
Luminosity change	72.06	82.66	70.41	70.37	72.95	81.44
Camouflage foreground object	54.14	73.51	70.20	76.30	82.23	89.97
Hybrid videos	85.69	72.20	80.69	79.75	82.63	91.50
Average	76.22	77.08	74.73	76.76	81.80	87.44

The results of our proposed system are copied from Table 2 for easy comparison in this table. Highest scores are highlighted in bold.

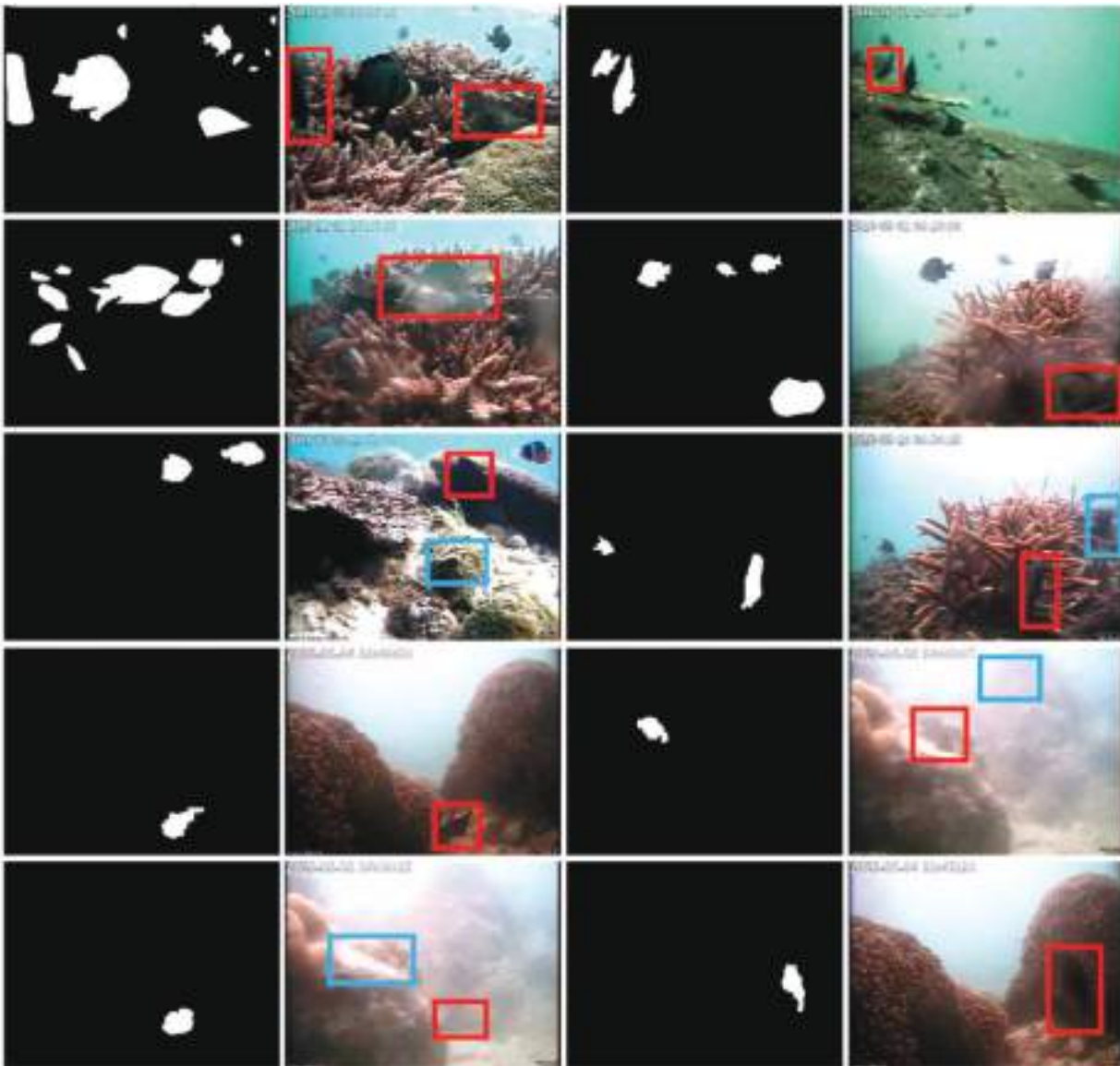


Figure 6. Examples of false detection of fish by all algorithms including our proposed hybrid system. Here, bounding boxes signify either miss detections of fish or false alarms. The black and white images are corresponding ground truths.

explanation for relatively worse performance of these approaches, as compared with the proposed system shown in Table 2, is an important observation that can be made by watching the videos

in the datasets. The fish in each frame may not necessary show motion and sometimes remain dormant for multiple frames, even though for most of the time they are swimming, making the

scenes dynamic. The GMM sometimes confuses the fish with the stationary profile as background, especially when the appearance of fish matches the background. Therefore, lack of motion information in video frames results in failure to detect fish by the GMM and optical flow. The R-CNN on the other hand is a tailored neural network used for object localization in the images (Ren *et al.*, 2017) and learns to capture fish-dependent information from stationary images.

Underwater fish detection in unconstrained environments is a challenging task as the main aim lies in segregating fish and ignoring non-fish entities in the entire video frame. Conventional machine learning and image processing algorithms are generally designed to detect the objects of interest in the datasets where they exhibit their distinct presence in the imagery, and hence are easier to segment out (Russakovsky *et al.*, 2015). In contrast, a high degree of confusion in separating fish with vibrant, diverse and variable non-fish objects in underwater videos results in a performance compromise for a standalone R-CNN with accuracy of 64.99 and 77.30% on the FCS and LCF-15 datasets, respectively. As mentioned earlier, many videos, especially in the FCS dataset, lacks textural and shape information of fish, a necessary ingredient to yield better performance by systems like standalone R-CNN. This problem is effectively solved by our proposed hybrid system using and learning the information from motion-sensitive and textural features. Figure 6 shows some results from the FCS and LCF-15 datasets where all algorithms including our proposed system failed to detect fish. These are the extreme cases of blurriness, camouflage, water murkiness, and unrecognizable orientation, texture, and shape of fish which either results in generating false alarms or miss detections. In these situations, it is extremely difficult to capture both motion-based and shape/texture-based features.

In the future, we aim to employ a unified deep architecture capable of processing the video sequences in real-time through rigorous optimization of our algorithm and better mathematical modelling. Such a setup will be applicable for fish detection as well as their species classification at the same time and, therefore, will be more suitable for effective fish fauna sampling. Furthermore, the accuracy of the system can be improved by tracking the paths of moving fish and having prior information of their movement in several frames. This step can improve the accuracy of detection in the video frames where the proposed approach fails to recognize fish due to extreme blurriness and the camouflage of the background. We plan to incorporate this processing step using recurrent neural networks (Gordon *et al.*, 2018) with temporal processing capability in videos.

Conclusions

In this article, we have presented an automatic method that employed deep R-CNN networks to detect and localize fish instances in unconstrained underwater videos that exhibit various degrees of scene complexity. The major contribution of this work is that it utilizes a hybrid approach involving GMM and optical flow outputs to combine motion sensitive input features with raw video frames carrying textural and shape information. This mixed data is used as input to a deep R-CNN to fine-tune the categorization of fish in the presence of non-fish entities in the video frame. This assisted in achieving state-of-the-art results for the fish detection task as confirmed by the comparative study. The proposed hybrid system requires relatively more computational resources as compared with the conventional computer vision

and machine learning techniques, but comes with the benefit of higher accuracy. However, with an advent of fast microprocessors and GPUs, complex mathematical operation involved in deep neural networks like CNN can be performed quickly, even making them suitable for tasks requiring near real-time processing. Therefore, combining the hybrid fish detection with other fish-related tasks like fish classification even using deep learning (Salman *et al.*, 2016) and tracking can be made possible in the pursuit of realizing fully automated systems for deployment in real world applications of fisheries. We believe that this research will help scientists related to fisheries in adopting automatic approaches for detection, classification and tracking of fish fauna in non-destructive sampling. Moreover, in the future, we aim to employ a unified deep architecture capable of processing the video sequences in real-time through rigorous optimization of our algorithm and better mathematical modelling. Such a setup will be applicable for fish detection as well as their species classification at the same time and therefore, will be more suitable for effective fish fauna sampling.

Acknowledgements

The authors acknowledge support from the Australian Research Council Grant LP110201008, and German Academic Exchange Service (DAAD) Project Grant 57243488 “FIBEVID”. The authors also acknowledge Nvidia Corporation, USA for their donation of graphics processing units (GPUs) under their GPU Grant Programme. Nvidia GPUs were used to carry out experiments in the work carried out in this article.

References

- Bengio, Y. 2009. Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2: 1–127.
- Beauchemin, S. S., and Barron, J. L. 1995. The computation of optical flow. *ACM Computing Surveys*, 27: 433–466.
- Brox, T., Bruhn, A., Papenber, N., and Weickert, J. 2004. High accuracy optical flow estimation based on theory for warping. *In Computer Vision-ECCV 2004. Lecture Notes in Computer Science*, 3024. Ed. by T. Pajdla and J. Matas. Springer, Berlin, Heidelberg.
- Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. 2009. ImageNet: a large-scale hierarchical image database. *IEEE CVPR-2009*, Miami, FL, USA, 248–255 pp.
- Fisher, R., Chen-Burger, Y.-H., Giordano, D., Hardman, L., and Lin, F.-P. (Eds) 2016. *Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data*. Intelligent Systems Reference Library, 104, Springer International Publishing. DOI: 10.1007/978-3-319-30208-9.
- Froese, R. 2006. Cube law, condition factor and weight length relationships: history, meta-analysis and recommendations. *Journal of Applied Ichthyology*, 22: 241–253.
- Gordon, D., Farhadi, A., and Fox, D. 2018. Re3: real-time recurrent regression networks for visual tracking of generic objects. *IEEE Robotics and Automation Letters*, 3: 788–795.
- Harvey, E. S., and Shortis, M. R. 1995. A system for stereo-video measurement of sub-tidal organisms. *Marine Technology Society Journal*, 29: 10–22.
- He, K., Zhang, X., Ren, S., and Sun, J. 2016. Deep residual learning for image recognition. *IEEE CVPR-2016*, Las Vegas, NV, USA, 770–778 pp.
- Hinton, G., and Salakhutdinov, R. 2006. Reducing the dimensionality of data with neural networks. *Science*, 313: 504–507.
- Hinton, G. E., Osindero, S., and Teh, Y.-W. 2006. A fast learning algorithm for deep belief nets. *Neural Computation*, 18: 1527–1554.

- Hsiao, Y., Chen, C., Lin, S., and Lin, F. 2014. Real-world underwater fish recognition and identification using sparse representation. *Ecological Informatics*, 23: 13–21.
- Huang, G., Liu, Z., and Weinberger, K. Q. 2017. Densely connected convolutional networks. *IEEE CVPR-2017*, Honolulu, HI, USA, 2261–2269 pp.
- Jennings, S., and Kaiser, M. J. 1998. The effects of fishing on marine ecosystems. *Advances in Marine Biology*, 34: 201–352.
- Larochelle, H., Bengio, Y., Louradour, J., and Lamblin, P. 2009. Exploring strategies for training deep neural networks. *Journal of Machine Learning Research*, 10: 1–40.
- LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. 1989. Backpropagation applied to handwritten zip code recognition. *Neural Computing*, 1: 541–551.
- LeCun, Y., Huang, F., and Bottou, L. 2004. Learning methods for generic object recognition with invariance to pose and lighting. *IEEE CVPR-2004*, Washington, DC, USA, 97–104 pp.
- LeCun, Y., Bengio, Y., and Hinton, G. 2015. Deep Learning. *Nature*, 521: 436–444.
- Lin, T. Y., RoyChowdhury, A., and Maji, S. 2015. Bilinear CNN models for fine-grained visual recognition. *IEEE ICCV-2015*, Santiago, Chile, 1449–1457 pp.
- McLaren, B. W., Langlois, T. J., Harvey, E. S., Shortland-Jones, H., and Stevens, R. 2015. A small no-take marine sanctuary provides consistent protection for small-bodied by-catch species, but not for large-bodied, high-risk species. *Journal of Experimental Marine Biology and Ecology*, 471: 153–163.
- Mika, S., Ratsch, G., Weston, J., Scholkopf, B., and Mullers, K. R. 1999. Fisher discriminant analysis with kernels. *IEEE International Workshop on Neural Networks for Signal Processing*, Madison, WI, USA, 41–48 pp.
- Moniruzzaman, M., Islam, S. M. S., Bennamoun, M., and Lavery, P. 2017. Deep learning on underwater marine object detection: a survey. *In Advanced Concepts for Intelligent Vision Systems. ACIVS 2017*. Ed. by J. Blanc-Talon, R. Penne, W. Philips, D. Popescu, and P. Scheunders. *Lecture Notes in Computer Science*, vol 10617, Springer, Cham.
- Palazzo, S., and Murabito, F. 2014. Fish species identification in real-life underwater images. *In 3rd ACM International Workshop on Multimedia Analysis for Ecological Data*, Orlando, Florida, pp. 13–18.
- Ren, S., He, K., Girshick, R., and Sun, J. 2017. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39: 1137–1149.
- Reynolds, D. A., and Rose, R. C. 1995. Robust text-independent speaker identification using Gaussian mixture speaker models. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 3: 72–83.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z. *et al.* 2015. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115: 211–252.
- Salman, A, Jalal, A., Shafait, F., Mian, A., Shortis, M., Seager, J., and Harvey, E. 2016. Fish species classification in unconstrained underwater environments based on deep learning. *Limnology and Oceanography: Methods*, 14: 570–585.
- Siddiqui, S. A., Salman, A., Malik, M. I., Shafait, F., Mian, A., Shortis, M. R., and Harvey, E. S. 2017. Automatic fish species classification in underwater videos: exploring pre-trained deep neural network models to compensate for limited labelled data. *ICES Journal of Marine Sciences*, 75: 374–389.
- Sheikh, Y., and Shah, M. 2005. Bayesian modelling of dynamic scenes for object detection. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 27: 1778–1792.
- Shortis, M., Harvey, E. S., and Abdo, D. 2009. A review of underwater stereo-image measurement for marine biology. *In Oceanography and Marine Biology: An Annual Review*. Ed. by R. N.Gibson, R.J.A. Atkinson, and J.D.M. Gordon. CRC Press, USA.
- Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv: 1409.1556*.
- Spampinato, C., Chen-Burger, Y., Nadarajan, G., and Fisher, R. B. 2008. Detecting, tracking and counting fish in low quality unconstrained underwater videos. *International Conference on Computer Vision Theory and Applications*, Funchal, Madeira, Portugal, 2: 514–519.
- Spampinato, C., Palazzo, S., and Kavasidis, I. 2014. A texton-based kernel density estimation approach for background modeling under extreme conditions. *International Journal of Computer Vision and Image Understanding*, 122: 74–83.
- Storbeck, F., and Daan, B. 2001. Fish species recognition using computer vision and a neural network. *Fisheries Research*, 51: 11–15.
- Strachan, N. J. C., and Kell, L. 1995. A potential method for the differentiation between haddock fish stocks by computer vision using canonical discriminant analysis. *ICES Journal of Marine Science*, 52: 145–149.
- Stauffer, C., and Grimson, W. E. L. 1999. Adaptive background mixture models for real-time tracking. *IEEE CVPR-1999*, Fort Collins, CO, USA, 2: 246–252.
- Sung, M., Yu, S., and Girdhar, Y. (2017). Vision based real-time fish detection using convolution neural network. *IEEE OCEAN-2017*, Aberdeen, UK, 1–6 pp.
- Szegedy, C., Ioffe, S., and Vanhoucke, V. 2016. Inception-v4, inception-resnet and the impact of residual connections on learning. *AAAI Conference on Artificial Intelligence*, Phoenix, AZ, USA, 4278–4284 pp.
- Tanzer, J., Phua, C., Lawrence, A., Gonzales, A., Roxburgh, T. and Gamblin P. (Eds) 2015. *Living Blue Planet Report. Species, Habitats and Human Well-Being*. WWF, Gland.
- Turk, M., and Pentland, A. 1991. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3: 71–86.
- Warren, D. H., and Strelow, E. R. 1985. *Electronic Spatial Sensing for the Blind: Contributions from Perception*. Springer. ISBN 90-247-2689-1.
- Yao, J., and Odobez, J. M. 2007. Multi-layer background subtraction based on color and texture. *IEEE CVPR-2007*, Minneapolis, MN, USA, 1–8 pp.
- Yong, X. 2013. Improved Gaussian mixture model in video motion detection. *Journal of Multimedia*, 8: 527–533.
- Zivkovic, Z., and Heijden, F. 2006. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters*, 27: 773–780.


Handling editor: Cigdem Beyan



Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Original Article

Automatic detection of Western rock lobster using synthetic data

Ammar Mahmood ^{1*}, Mohammed Bennamoun¹, Senjian An², Ferdous Sohel³, Farid Boussaid¹, Renae Hovey¹, and Gary Kendrick¹

¹The University of Western Australia, WA 6009, Australia

²Curtin University, WA 6845, Australia

³Murdoch University, WA 6150, Australia

*Corresponding author: tel: +61424690778; e-mail: ammarmahmood@live.com.

Mahmood, A., Bennamoun, M., An, S., Sohel, F., Boussaid, F., Hovey, R., and Kendrick, G. Automatic detection of Western rock lobster using synthetic data. – ICES Journal of Marine Science, 77: 1308–1317.

Received 23 April 2019; revised 19 October 2019; accepted 22 October 2019; advance access publication 22 November 2019.

Underwater imaging is being extensively used for monitoring the abundance of lobster species and their biodiversity in their local habitats. However, manual assessment of these images requires a huge amount of human effort. In this article, we propose to automate the process of lobster detection using a deep learning technique. A major obstacle in deploying such an automatic framework for the localization of lobsters in diverse environments is the lack of large annotated training datasets. Generating synthetic datasets to train these object detection models has become a popular approach. However, the current synthetic data generation frameworks rely on automatic segmentation of objects of interest, which becomes difficult when the objects have a complex shape, such as lobster. To overcome this limitation, we propose an approach to synthetically generate parts of the lobster. To handle the variability of real-world images, these parts were inserted into a set of diverse background marine images to generate a large synthetic dataset. A state-of-the-art object detector was trained using this synthetic parts dataset and tested on the challenging task of Western rock lobster detection in West Australian seas. To the best of our knowledge, this is the first automatic lobster detection technique for partially visible and occluded lobsters.

Keywords: deep learning, lobster detection, marine science, synthetic data, underwater images

Introduction

Lobsters are economically important as they are highly prized seafood. They are considered as one of the most profitable commodities in coastal areas they populate. The Western rock lobster, *Panulius cygnus*, is one of the Australia's most significant natural marine resources. The industry is worth over \$300 million Australian dollars a year. From a fisheries perspective, detailed knowledge on lobster abundance and key habitat associations within the deeper water fishing grounds is needed for the spatial management of the species, to increase the capacity to assess

effects of fishing and ultimately enhance the sustainability of the Western rock lobster fishery. However, it is logistically difficult to gather the required data as lobsters are cryptic animals that move from their shelters at night and at water depths that are beyond scuba capabilities. Pots and traps are the traditional methods for sampling lobsters (Bellchambers *et al.*, 2010). However, these methods are heavily biased as baits are used to lure animals with no record of habitat. Underwater images collected through autonomous underwater vehicles (AUVs) are a potential technique that will allow for the sampling of lobsters in their natural

foraging habitats. AUVs have already been used successfully for other commercial or ecologically important species (Tolimieri *et al.*, 2008; Grothues *et al.*, 2008; Williams *et al.*, 2009, 2010). AUVs are capable of collecting hundreds of thousands of images during a single survey campaign. As a result, processing the acquired massive data has become the new challenge (Mahmood *et al.*, 2018).

For automatic detection using deep learning-based lobster detector, a large amount of annotated training data is required to optimize the large number of parameters (Girshick *et al.*, 2014; Redmon *et al.*, 2016; Ohn-Bar and Trivedi, 2017; Wei *et al.*, 2018). Generating large amounts of training data for the aforementioned process is time-consuming and resource hungry. Moreover, the number of images containing the objects of interest is very limited. This is quite often the case for underwater species, such as lobster. This gets more problematic when the labels are difficult to specify manually without expert knowledge. Another important limiting factor is the reduced generalization capability of the learned models across diverse backgrounds and unseen environments.

To overcome these limitations, the generation of synthetic data has been investigated (Dwibedi *et al.*, 2017; Tobin *et al.*, 2017; Tremblay *et al.*, 2018). However, all previously proposed synthetic data approaches for object detection segment the whole objects of interest and use them to generate synthetic data. This approach is simple when the objects of interest have a simple shape such as most objects in indoor scenes. For objects with a complex body, such as living organisms, the task of manually or automatically segmenting the target objects for synthetic data generation is far more complex. To solve this challenge, we propose to generate synthetic data for parts of the lobster rather than the whole lobster. Our proposed method is especially useful when the object of interest is occluded and only partially visible. Moreover, in some complex scenes, an object may only be detected by its most prominent part(s). In this article, we demonstrate the proposed synthetic parts data (SPD) generation for the challenging task of lobster detection.

Automatic detection of lobsters in a complex underwater background poses an interesting computer vision problem in the object localization domain. The amount of collected lobster images is not enough to train a deep learning-based object detector. Relying on synthetic parts can be a promising approach for lobster detection because, in most acquired images, the lobster would be occluded and only its antennae are visible (Figure 2). This observation helps experts localize lobsters in many images in their manual annotation process.

In this article, we propose a synthetic data generation approach for the task of lobster detection. Because a lobster is an object with a complex shape, we explore the idea of generating synthetic parts instead of the whole body to achieve a higher performance. We hypothesize that the synthetic data generated using the most prominent body part (lobster's antennae in this scenario) will achieve the highest detection accuracy. In particular, we are interested in answering the following questions: (i) Can SPD achieve better results on real lobster data? (ii) How much synthetic data used to train an object detector is sufficient? (iii) Can we improve the baseline performance by only using a small amount of real data with additional synthetic data? (iv) How can we best bridge the gap between synthetic and real data?

The main contributions of this article are as follows:

- (i) We report the first automated approach for lobster detection.
- (ii) We propose to use a novel SPD approach for generating synthetic data for the lobster, a complex bodied object.
- (iii) We use the SPD to train and apply the state-of-the-art, YOLOv3 (Redmon and Farhadi, 2018), object detection framework for lobster detection.
- (iv) Based on our results and observations, we offer insights regarding the optimal amount of SPD and real data required for achieving a higher object detection rate for lobster detection.

The rest of the article is organized as follows. “Related work” section summarizes relevant previous research. “Proposed method” section explains our proposed method in detail. “Experiments” section describes our experimental work and reports the results. “Results” section concludes this article.

Related work

Lobster detection

Traditional methods for detecting lobsters use pots and traps (Bellchambers *et al.*, 2010). These methods, however, lack information on true habitats of lobsters as baits are used to lure them outside their habitats. Another semi-automatic approach relies on underwater videos to detect and count lobsters (Correia *et al.*, 2007). Their algorithm was based on the observation that lobsters tend to be in the bright area of images. The following three visual features were used in their work to track lobsters: intensity, motion and edges. This work was extended by Lau *et al.* (2008) to further enhance the lobster tracking algorithm.

Tan *et al.* (2018) proposed a comprehensive lobster and burrow detection and tracking algorithm based on underwater videos. This method also relies on detecting lobster from the brighter regions of the images and uses features based on curvature, local intensity contrast, aspect ratio, and orientation of lobster. However, their method is not generic as the images used in their experiments do not have a complex background. Most of the time, the lobster is not occluded and is fully visible, which is not the case with our dataset.

As seen, there is very limited research conducted on the automatic detection of lobsters. However, there are also works on detecting species, which are close to lobster, such as crabs and prawns. Pedersen *et al.* (2019) introduced Brackish dataset, which has classes, such as crab, shrimp, jellyfish, and fish. They have reported baseline results on this dataset using YOLOv2 and YOLOv3 frameworks. In a similar study (Wang *et al.*, 2018), crab detection is performed by localizing knuckles instead of the whole crab using a convolutional neural network (CNN). However, this method is tested for images of crabs on a meat picking machine, a relatively simpler scenario as compared with the complex backgrounds encountered in the ocean in our case.

Synthetic data

In recent years, the use of synthetic data has become popular in training and testing deep networks for various object detection and segmentation tasks. Gupta *et al.* (2016) generated synthetic images by placing text randomly onto the images for the task of

text detection in the wild. The resulting synthetic dataset is then used to train a deep learning-based object detector. Our proposed method is similar to this approach in the context that we also place synthetic parts randomly onto diverse background images. However, our application scenario is more complex than text detection in the wild. The work of [Georgakis et al. \(2017\)](#) shows that carefully placing synthetic objects in the indoor scenes can improve the performance of object detectors significantly. They show that the geometric context and semantics of scenes are more important than just randomly inserting objects anywhere in the indoor scenes. However, the underwater scenes do not have geometric context, such as tables and counter-tops in the kitchen scenes used in [Georgakis et al. \(2017\)](#).

[Tobin et al. \(2017\)](#) introduced the concept of domain randomization (DR) to close the gap between real and synthetic data. They argue if they generate synthetic data with a large amount of variations, the model will treat the real data as just another variation. They used DR to train a neural network to estimate the positions of various objects with respect to a robot. Our proposed method also uses the concept of DR in a way such that we have used very diverse background images to generate our synthetic data.

Similar to DR, another approach is to segment real objects from a source dataset and add them onto background images from a different dataset to generate synthetic data for training deep networks ([Dwibedi et al., 2017](#)). One limitation of this method is the automatic segmentation step, which becomes complicated when objects of interest have a complex shape as in the case of lobsters.

[Tremblay et al. \(2018\)](#) demonstrate that DR is an effective technique to bridge the gap between synthetic and real images. They claim that the synthetic data do not need to be photorealistic as patch-level realism is sufficient for training region proposal-based object detectors. They also introduce the concept of adding noise in the form of flying distractor objects in the synthetic images to make the deep networks more robust and accurate. We have used the concept of patch-level realism in our proposed method since our generated synthetic data are not photorealistic at a global level.

All the aforementioned research focus on using whole objects to generate synthetic data. To the best of our knowledge, no other previous studies generate synthetic data for parts of objects. In this article, we also report the first application of synthetic data to living organisms in general and Western rock lobsters in particular.

Proposed method

This section introduces the proposed method for lobster detection. First, we introduce our synthetic data generation approach for lobster parts. Second, we introduce the *You Only Look Once* (YOLO) object detector used in our experiments. [Figure 1](#) illustrates our proposed method for generating SPD for lobster detection.

Synthetic data generation

Unlike the synthetic objects generated in previous studies, such as kitchen objects ([Dwibedi et al., 2017](#)) and cars ([Tremblay et al., 2018](#)), our proposed method is focused on generating SPD for objects with complex shapes. The first step in synthetic data generation is to segment the object of interest from real images either manually or automatically. Segmenting objects with complex

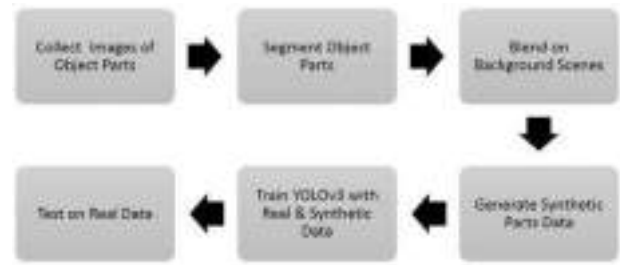


Figure 1. The block diagram of our proposed framework.

shapes is a tediously time-consuming task as compared with segmenting objects used in previous works ([Dwibedi et al., 2017](#)). For example, segmenting a lobster from real images is a far more difficult task than segmenting a car or a bottle. Hence, we propose to generate synthetic data for a prominent part instead of the whole object. Another benefit of our proposed approach is that in most of the real images, the object of interest is occluded and not fully visible. Moreover, in the case of lobster images, the underwater background is very complex and it is difficult to locate the lobster if we are looking for its whole body. In many of the real images, only the antennae are visible and the human annotator used the antennae to locate the lobster. This observation led us to only generate the synthetic parts of the lobster instead of its whole body. The main steps of our proposed method are explained below.

Collecting images

In this section, we describe how we collected the lobster parts from real images and the background images to paste the object parts on.

Images of lobster parts: We used the lobster images present in the dataset to segment the parts by hand. The lobster dataset consists of 237 ground annotated images collected from Western Australia (WA). This dataset was captured using the same sensor as that of Benthos15 dataset ([Bewley et al., 2015](#)). A stereo camera pair of 1.4 MP was used to capture both datasets from a distance of ~ 2 m. The camera field of view is $\sim 42^\circ \times 34^\circ$, and the images have a resolution of 1360×1024 pixels. The images from the lobster dataset can have multiple lobsters or no lobsters at all. The lobsters are usually observed at different scales. Most of the lobsters in this dataset range between 90 and 120 mm in length. Bounding box annotations are drawn on these images wherever a lobster is encountered whether it is occluded or not. To provide an appreciation of the complexity of this dataset, [Figure 2](#) depicts two sample images of a lobster in an underwater environment. [Figure 2a](#) shows a lobster in a relatively easy to detect background. However, in [Figure 2b](#), it is difficult to locate the lobster due to the presence of occlusions, which make it a challenging object detection task.

We selected sample instances for the following two types of body parts for lobster: (i) antennae of lobster and (ii) body of lobster.

Background images of Benthos15 dataset: We used images from the Benthos15 dataset as background scenes for pasting lobster parts. The Benthos15 dataset ([Bewley et al., 2015](#)) consists of an expert-annotated set of underwater images captured by an AUV deployed around Australia. The whole dataset contains 9874 distinct images collected at different depths from nine sites around Australia over the past few years. We used only a subset

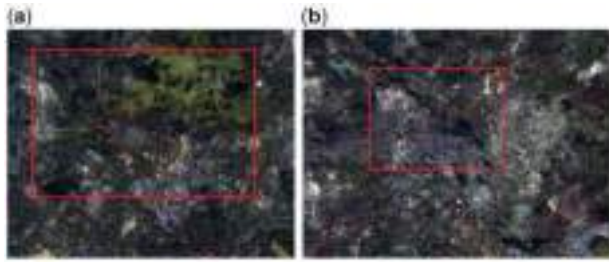


Figure 2. Example images from the lobster dataset: (a) easy case and (b) difficult case.

of this dataset with images taken from WA to generate our synthetic images. Images from year 2012 were used to generate synthetic data for the lobster's antennae, and images from year 2013 were used to generate synthetic data for the lobster's body. We have selected the Benthos15 dataset as a source of our background images because of the proximity of the data sites to the lobster habitats, which are considered important lobster fishing grounds. In addition, this dataset offers a vast variety of marine images, which helps our object detector become more robust to the diversity and complexity of underwater scenes.

Lobster part segmentation: After collecting images containing lobsters, the antennae and the body parts were segmented from the background. This step was performed manually using the Ratsnake annotation tool. Ratsnake is a publicly available software to generate segmentation masks for images. We used the default settings of Ratsnake to segment lobster parts. The aforementioned steps are illustrated in Figure 3.

Adding synthetic parts to images

After segmenting the lobster antennae and body from the input images, we inserted them into real and diverse background images from the Benthos15 dataset. Naively pasting the lobster parts on the background images without any blending can lead to boundary artefacts, which can affect the object detectors. These boundary artefacts may seem subtle but they are known to reduce the performance of object detection frameworks (Dwivedi *et al.*, 2017). Moreover, given that state-of-the-art object detectors largely depend on local features and region proposals (Girshick *et al.*, 2014), training these networks on input images with boundary artifacts severely reduces their performance. To overcome this limitation, we have added a blending step to paste the lobster parts onto the background images. In our proposed method, we used a Poisson blending scheme (Pérez *et al.*, 2003) to smooth the edges of the pasted part and to also add illumination variations. Specifically, we used the Poisson blending with the seamless cloning and colour shifting options (Pérez *et al.*, 2003). Seamless cloning option takes care of merging the boundaries of pasted object with the background. The colour shift setting ensures that there are no unnatural illumination changes, such as a dark lobster part when pasted in a brighter region should appear brighter. This step contributes notably towards making the object detector less sensitive to the induced boundary artifacts and thus more robust when detecting the shape of the blended object.

We also utilized DR to make the synthetic data as diverse and random as possible. The following three factors of our proposed method contributed towards DR: (i) using diverse images with no repetitions from the Benthos15 dataset as background of the

synthetic images; (ii) using Poisson blending for creating illumination variations in the synthetic images; and (iii) by randomly placing, scaling and rotating the lobster parts for synthetic data generation. This step increases the robustness of the object detector trained using the SPD. A few randomly chosen synthetic images of lobster antennae and body are shown in Figures 4 and 5, respectively. These examples illustrate the complexity and diversity of the underwater scenes.

Object detection model

We used the third version (YOLOv3; Redmon and Farhadi, 2018) of the real-time object detection model YOLO (Redmon *et al.*, 2016) to test the synthetic part data generated by our proposed method. We have selected YOLOv3 over other object detectors because it runs significantly faster than other detection methods with comparable performance. Moreover, the selection of a faster object detection model will ensure real-time lobster detection. Figure 6 illustrates the complete proposed framework.

YOLOv3 employs a variant of a Darknet (Redmon and Farhadi, 2018) as a feature extractor. This variant has 53 layers and is pre-trained on the ImageNet dataset. YOLOv3 uses the following three scales for the detection of objects: 1/8, 1/16, and 1/32 of the input image size. It uses nine anchor boxes, three for each scale. These anchor boxes are generated using the K-means clustering technique. We have used the same training protocol for YOLOv3 as described in Redmon and Farhadi (2018). We did not use any pre-processing for input images. Images of size 892×892 pixels were used as input to the YOLOv3. We trained each model for 50 epochs using an initial learning rate of 0.001 with a mini-batch size of 8 and no dropout. The learning rate was further divided by 10 after every 20 epochs. The momentum and weight decay were set to 0.9 and 0.0005, respectively, throughout the training. We used the Adam solver (Kingma and Ba, 2014) for optimization. We also employed early stopping for situations where the validation loss stopped decreasing.

Experiments

In this section, we compare the performance of our synthetic part data approach against the baseline performance achieved using only real data. We have used 80:20 training to testing split on our lobster dataset. Throughout our experiments, the test set consists of 50 images of the total 237 images unless stated otherwise. No image or part of an image from the test set was used in synthetic data generation or training the object detector. The test set was only reserved for evaluation of the proposed method. Regarding our lobster dataset, it only contains 237 images. These images were curated from a very large dataset with tens of thousands of images collected during an AUV survey. Lobsters are very difficult to locate in their habitats. Moreover, there are no publicly available lobster datasets. For each experiment, 20% of images in the training set are used only for validation to optimize the experiment parameters. For *synthetic parts dataset*, we have generated 500 images with only the lobster antennae and another 500 images with lobster body only. It is important to note that no background image was used twice in the 1000 synthetic parts images. Moreover, we have applied scaling and rotation randomly while randomly placing the lobster parts on background images to generate the synthetic images. This step adds to the diversity of the synthetic dataset. For the YOLOv3 object detector, we have used the weights from a model pre-trained on the PASCAL VOC

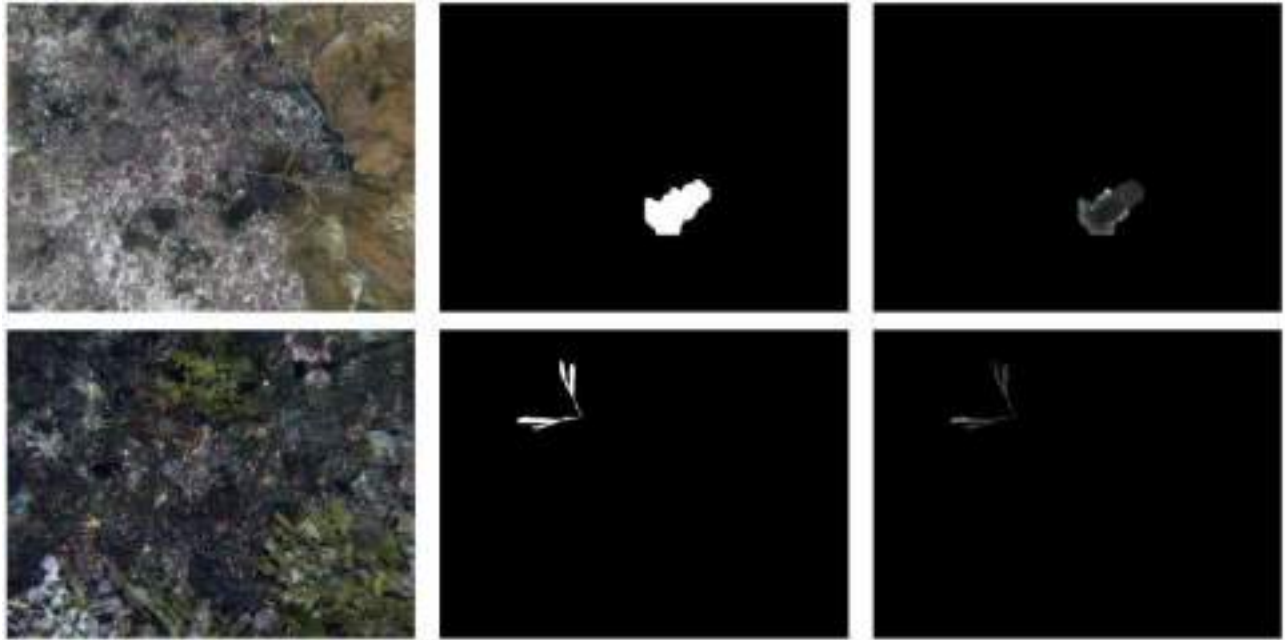


Figure 3. Collecting lobster body parts from real images. Left: real images with lobsters. Middle: segmented lobster parts. Right: segmented parts overlaid on real images.

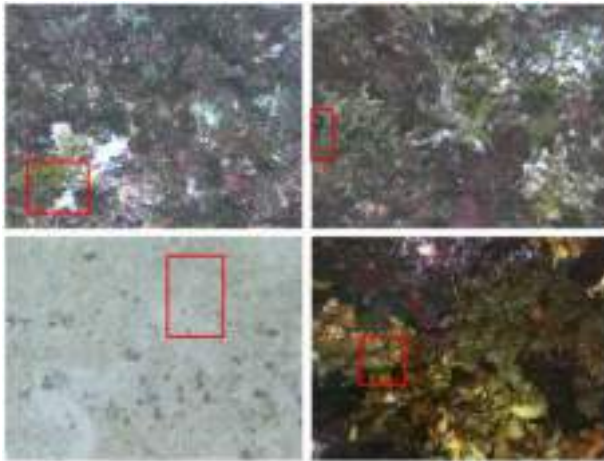


Figure 4. Randomly chosen samples from our synthetic lobster antennae images.

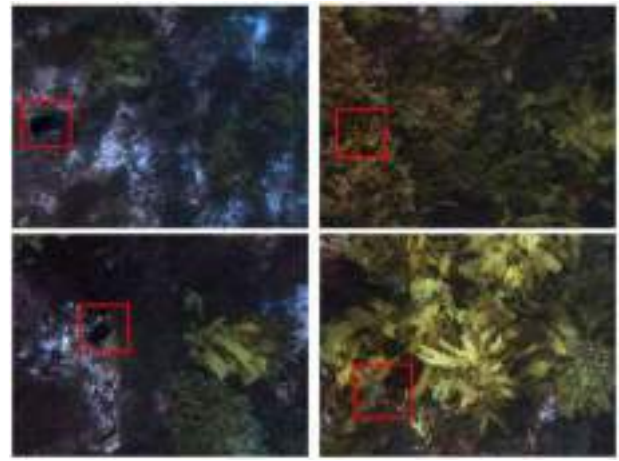


Figure 5. Randomly chosen samples from our synthetic lobster body images.

dataset because the lobster dataset is not large enough to train this model from scratch. We did not freeze the weights of any layer of this pre-trained model while fine-tuning. We will present the results of the following three studies in this section:

- (i) entire lobster dataset and entire synthetic parts dataset;
- (ii) entire synthetic parts dataset but varying number of real images from the lobster dataset; and
- (iii) entire lobster dataset but varying number of synthetic images from the synthetic parts dataset.

In our experiments, we report the mean average precision (mAP) at the intersection over union of 0.5 for the lobster detection task.

The mAP is defined as the mean precision at the set of 11 equally spaced recall values (i.e. $[0, 0.1, 0.2, \dots, 1.0]$) obtained from the precision–recall curve of the model’s detection output. We also use a non-maximum suppression (NMS) threshold value of 0.5 to get rid of overlapping detections before the mAP is calculated. The multiple detections usually occur when a full lobster is present in a test image (both antennae and body). The lobster detector is trained on synthetic body and antennae data as well as real images and some of them has a full lobster in them. As a result, the object detector produces multiple boxes for a single lobster. The imposed threshold on NMS ensures that these multiple detections are not taken into consideration when calculating mAP.

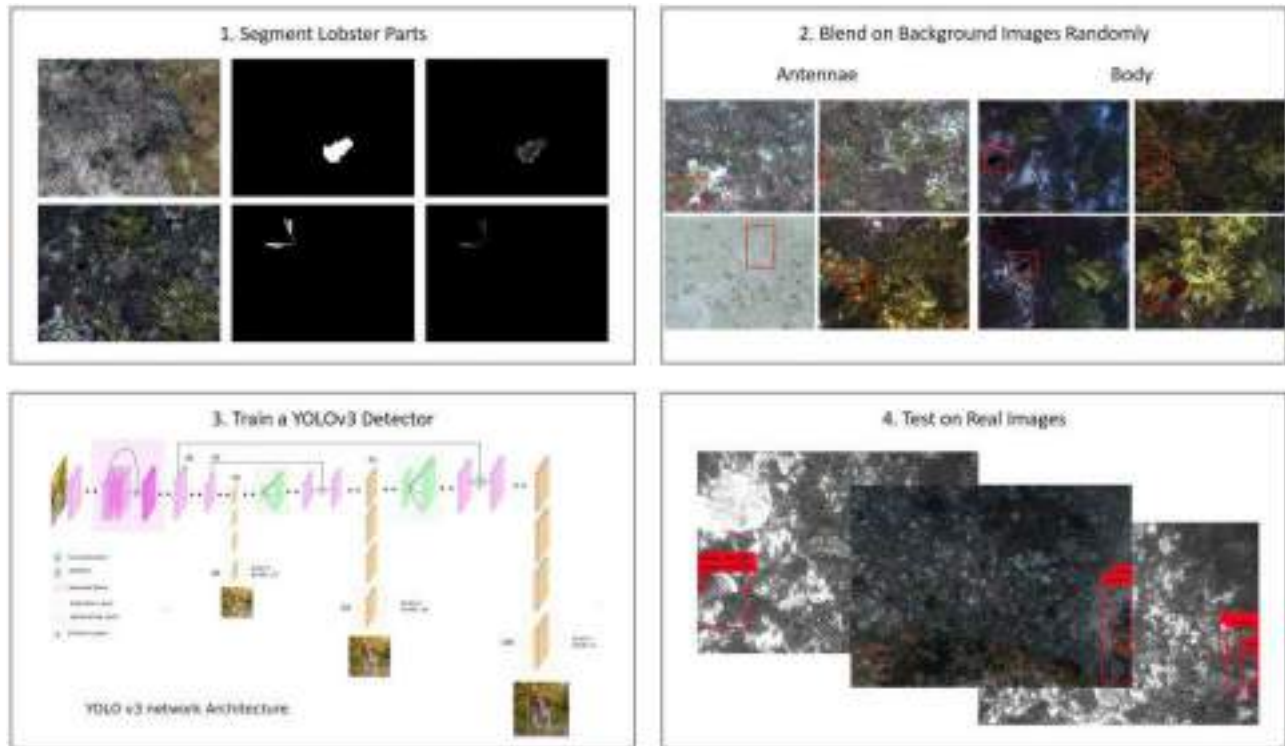


Figure 6. Synthetic parts data generation for lobster detection: (1) segmenting lobster parts, (2) blending on background images of Benthos15 dataset, (3) training a YOLOv3 object detector, and (4) testing on real images.

Evaluation on entire datasets

To test the effect of SPD on lobster detection, we conducted three experiments using the entire lobster and synthetic parts datasets. In the *Real Only* experiment, we trained the YOLOv3 model with 187 images in the training set. This experiment serves as baseline for all the remaining experiments. For *Synthetic Only* experiment, no real image was used for training. The associated training set consists of 1000 images (500 images from synthetic antennae and 500 images from synthetic body). For the third experiment, *Real + Synthetic*, the training set consists of 187 real images and the 1000 synthetic images. The same test set was used for all conducted experiments.

Varying real data

We conduct three experiments where 0, 50 and 100 real images from the lobster dataset are used to train the object detector along with the entire synthetic parts dataset. Entire synthetic parts dataset is also included in the training data. Each model is then tested on the following two sets of images: the standard test set containing 50 images and a second test set formed by using all the real images that were not used for training in the lobster dataset.

Varying SPD

We investigated the effect of varying the SPD combined with the entire lobster dataset. We conducted multiple experiments with a varying amount of synthetic parts images and real images.

For each experiment, synthetic antennae images and synthetic body images contribute equally in forming the synthetic image set used to train the YOLOv3 detector.

Results

Evaluation on entire datasets

Table 1 details the training and test sets for these three experiments and also reports the achieved mAP. The *Real Only* experiment achieved an mAP of 21.0, which forms our baseline performance. We augmented the real dataset by adding flips, crops, and rotated images for each real image using a random sampling strategy. The improvement achieved by the data augmentation step was marginal (mAP increased by 1.8) as compared with the performance achieved by adding synthetic data as demonstrated below. However, we did not add this step to the remaining experiments because we wanted to investigate the optimal relationship between the amount of real and synthetic images to achieve high performance gains. Adding a data augmentation step to the real images would not have given us a clear insight into this investigation. Therefore, we skipped this step for the remaining experiments. The mAP reduces drastically to 2.1, when there is no real data used in training the object detector. This result highlights the fact that synthetic data cannot generalize to an unseen set of real test data. However, the *Real + Synthetic* experiment achieves an mAP of 46.9, which is an improvement of 25.9 over the baseline mAP. This result shows that synthetic data can improve lobster detection by a significant margin. When the model is trained only on synthetic images, it only learns to detect

Table 1. Performance evaluation on the entire lobster dataset and the entire synthetic parts dataset.

Experiment	Training data			mAP
	Real images	Synthetic body images	Synthetic antennae images	
DPM v5 (Girshick et al., 2012): Real Only	187	0	0	12.1
DPM v5 (Girshick et al., 2012): Real + Synthetic	187	500	500	34.9
Regionlets (Wang et al., 2013): Real Only	187	0	0	16.3
Regionlets (Wang et al., 2013): Real + Synthetic	187	500	500	38.1
R-CNN (Girshick et al., 2014): Real Only	187	0	0	18.0
R-CNN (Girshick et al., 2014): Real + Synthetic	187	500	500	41.5
SPD: Real Only	187	0	0	21.0
SPD: Real Only (aug)	187	0	0	22.8
SPD: Synthetic Only	0	500	500	2.1
SPD: Real + Synthetic	187	500	500	46.9
SPD: Real + Antennae	187	0	500	48.7
SPD: Real + Body	187	500	0	40.9

partial lobster parts, such as antennae or lower body. When this model is tested on an image with a full lobster present in it, it will produce partial bounding boxes and will result in a lower mAP value. On the other hand, when synthetic dataset is used with real data to train the model, it adds to the semantic information required to detect partially occluded lobsters and results in a higher mAP value.

Figure 7 shows some example images where the model *Real + Synthetic* was successful in detecting the lobster whereas the model *Real Only* failed. Figure 8 shows examples of test images where all of the models failed to detect a lobster. It can be observed that these example images are quite challenging and the lobster is located either at the edge of the image or in shadows.

We also conducted two more experiments to compare the effect of different synthetic parts on the model's performance. In *Real + Antennae* experiment, only the synthetic images with the lobster antennae are used for training. Similarly, in *Real + Body* experiment, the model is trained on the real images and the synthetic body images only. The results in Table 1 show that the model trained on synthetic antennae images outperforms the model trained on synthetic body data by a significant margin of 7.8 in mAP. This result confirms our initial hypothesis that the synthetic data generated for the object's most prominent part (lobster's antennae in this case as shown in Figure 2a) achieves the highest detection performance. It is interesting to note that the model trained only on synthetic antennae images has also outperformed the model trained on all the synthetic data by a nominal margin of 1.8. This performance improvement can be explained by the fact that only the lobster's antennae are visible in some of the images. In most cases, the lobster's body is either occluded or masked by shadows and is difficult to localize in the complex background, even by expert human annotators. Hence, lobster's antennae are a distinguished landmark, which helps human annotators to locate a lobster with ease. The performance improvement achieved by the antennae only SPD is attributed to this observation.

We also evaluated the performance of our approach against traditional computer vision-based object detection methods for the lobster detection task, including the latest variant of deformable parts method (DPM) for this experiment (Girshick et al., 2012). This method was one of state-of-the-art methods before deep learning method became popular (Girshick et al., 2014). DPM is based on the histogram of oriented gradients (HOG)

features and it uses latent support vector machines (L-SVM) for classification. Our experimental results show that our proposed method outperforms DPM by a large margin for both experiments: *Real Only* and *Real + Synthetic* (Table 1). We also tested the Regionlets object detection method of Wang et al. (2013) for lobster detection. This method is based on candidate region selection (called Regionlets), and it uses HOG and Local Binary Patterns as descriptors followed by a cascaded boosting classifier, which is learned by selecting the most discriminative Regionlets. Table 1 shows that our proposed method achieves a much higher mAP as compared with Regionlets. To highlight the importance of the choice of the object detector, we conducted experiments using the R-CNN object detection framework (Girshick et al., 2014). We used a model of R-CNN pre-trained on the PASCAL VOC dataset. The momentum and weight decay were set to 0.9 and 0.0005, respectively, throughout the training. The initial learning rate was set to 0.001 and was further reduced by a factor of 10 when the validation loss stopped decreasing. The remaining hyper-parameters were set to default values as detailed in Girshick et al. (2014). The reported results show that the more advanced YOLOv3 object detector outperforms the R-CNN object detector in each of the experiment.

Varying real data

Table 2 reports the mAP of experiments of this study where we vary the number of real images in the training set. Table 2 shows that the mAP increases as we increase the number of real images to train the object detector. It also shows that we can achieve a higher mAP than the baseline mAP of 21.0 by using just the 100 real images from the lobster dataset. The mAP values reported for each experiment are marginally similar for the two test sets indicating that our standard test set, although small, is a good representation of the entire lobster dataset. Moreover, the results of Table 2 also indicate that our proposed approach reduces the amount of real annotated training data required to achieve a higher than baseline performance by a significant margin (100 real images instead of 187 required previously).

Varying SPD

Table 3 reports the mAP of experiments of this study where we vary the number of synthetic images in the training set. For 187 real images, the highest mAP is achieved by using 500 synthetic

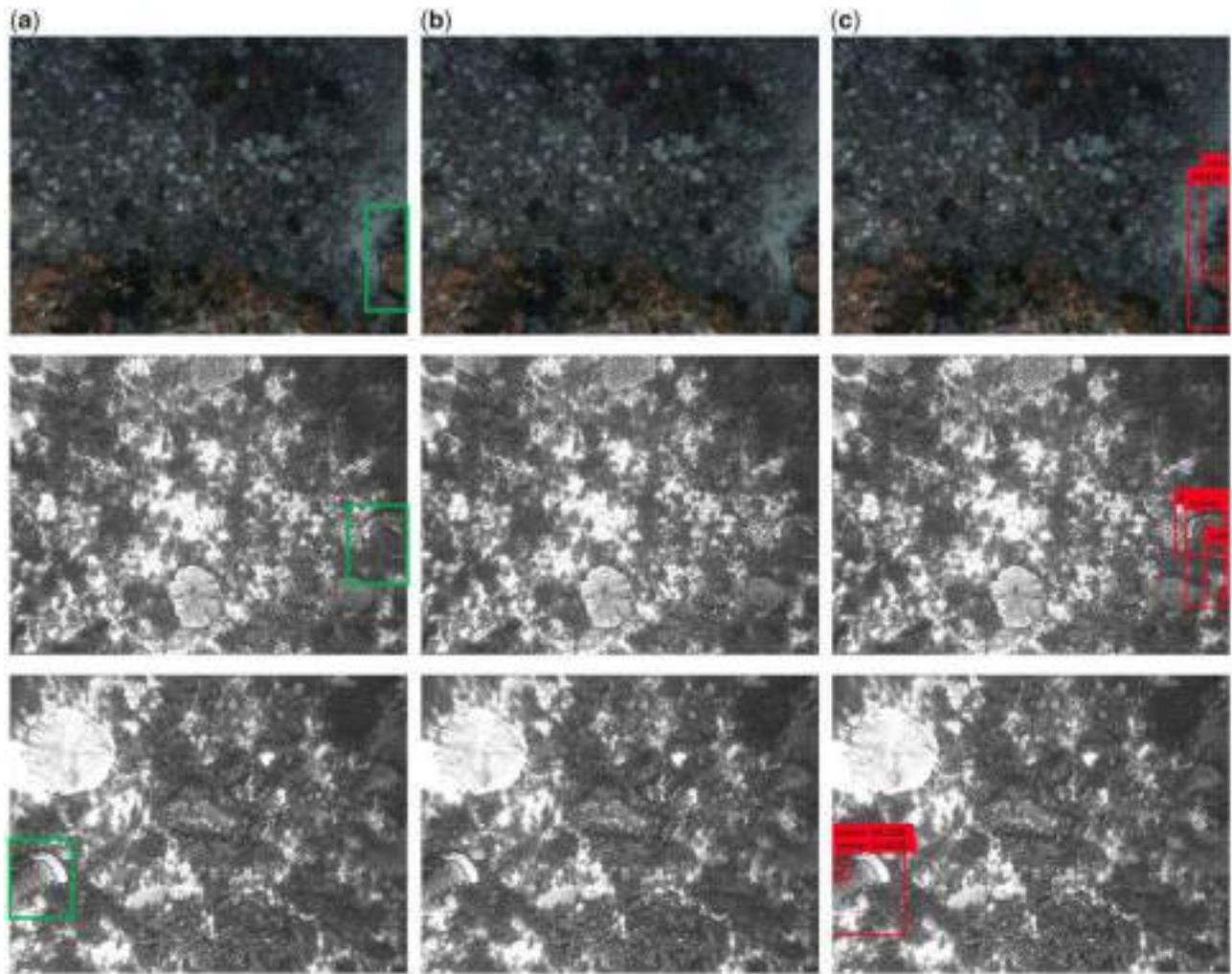


Figure 7. Detection examples for different models: (a) ground truth, (b) *Real Only*, and (c) *Real + Synthetic*.

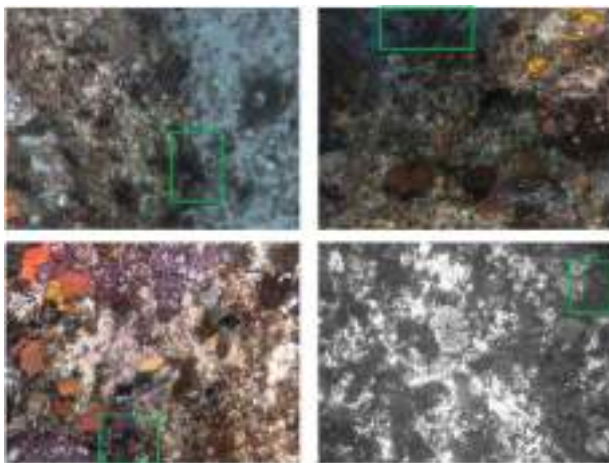


Figure 8. Missed detections on the lobster dataset.

images in the training set (250 each from antennae and body images). The mAP achieved by including 100 and 250 synthetic images in the training set is also comparable to the highest mAP achieved. However, the mAP reduces from 51.1 to 46.9, a drop of

Table 2. Performance evaluation on the entire lobster dataset by varying the amount of real data from the lobster dataset.

Experiment	Real images	Test data	mAP
Synthetic Only	0	50	2.1
	0	237	2.2
50 Real + Synthetic	50	50	11.9
	50	187	12.2
100 Real + Synthetic	100	50	26.5
	100	137	26.4
187 Real + Synthetic	187	50	46.9

4.2, when we increase the amount of synthetic data from 500 to 1000 images. A similar trend is also observed for the experiment with 50 real images as shown in Table 3. We further extended this analysis by selecting the best models for each experiment: (i) 187 real images with 500 synthetic images and (ii) 50 real images with 250 synthetic images for sensitivity analysis. We applied these two models five times using random combinations of 500 and 250 synthetic images, respectively. The resulting mAP values are shown in Figure 9 using a boxplot to demonstrate the internal sensitivity of each model. We observed that there was a greater

Table 3. Performance evaluation on the entire lobster dataset by varying the amount of synthetic data from the synthetic parts dataset.

Synthetic images	mAP	
	187 real images	50 real images
1 000	46.9	11.9
500	51.1	27.8
250	50.7	33.4
100	49.1	29.7
0	21.0	17.1

For each experiment, we have used the same test set.

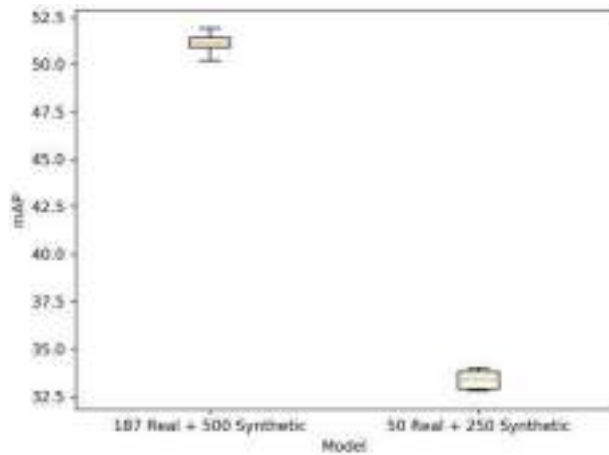


Figure 9. Boxplot of mAP for the best models.

variability in mAP values of the second model, which had fewer real images in the training data compared with the first one. Adding more real images to the training data, thus, makes the resulting model more robust to the addition of synthetic data. Moreover, we conducted these experiments by keeping the amount of real images equal to synthetic images in each mini-batch for each training epoch. We achieved similar results with minor improvement in the mAP values (mAP within ± 0.9 of the corresponding reported values of Table 3).

The results in Table 3 suggest that adding too much synthetic data in the training set can be detrimental to a model's performance. Previous studies on synthetic data have not reported any similar results where increasing the amount of synthetic data after a certain limit starts to reduce the performance. This performance drop can be attributed to the fact that when trained with excessive synthetic data, the object detector gets biased towards synthetic geometric contexts and results in detection errors when tested on real images.

Conclusion

We have demonstrated that the SPD approach is an effective technique to generate synthetic data for the automatic detection of Western rock lobster. Using the synthetic parts dataset, we have trained a YOLOv3 object detector and used it to detect lobsters in challenging underwater images. Our results show a significant performance improvement over the baseline method, which only uses real images for training the network. We have also demonstrated that using synthetic data of the most

prominent part of an object can further improve the performance of the object detector as compared with the synthetic data from other parts. We also show that the synthetic data along with a small portion of real data can achieve performance higher than the baseline. Our proposed method is a promising step towards the automation of marine species detection with complex body shapes, partially accessible local environments, and limited availability of training datasets.

Acknowledgements

This research was partially supported by Australian Research Council Grants (DP150104251 and DE120102960) and the Integrated Marine Observing System through the Department of Innovation, Industry, Science and Research, National Collaborative Research Infrastructure Scheme. The authors also acknowledge NVIDIA for providing a Titan-X GPU for the experiments involved in this research.

References

- Bellchambers, L. M., Evans, S. N., and Meeuwig, J. J. 2010. Abundance and size of western rock lobster (*Panulirus cygnus*) as a function of benthic habitat: implications for ecosystem-based fisheries management. *Marine and Freshwater Research*, 61: 279–287.
- Bewley, M., Friedman, A., Ferrari, R., Hill, N., Hovey, R., Barrett, N., Pizarro, O., et al. 2015. Australian sea-floor survey data, with images and expert annotations. *Scientific Data*, 2.
- Correia, P. L., Lau, P. Y., Fonseca, P., and Campos, A. 2007. Underwater video analysis for Norway lobster stock quantification using multiple visual attention features. *In 2007 15th European Signal Processing Conference*, pp. 1764–1768. IEEE.
- Dwivedi, D., Misra, I., and Hebert, M. 2017. Cut, paste and learn: surprisingly easy synthesis for instance detection. *In The IEEE International Conference on Computer Vision (ICCV)*.
- Georgakis, G., Mousavian, A., Berg, A. C., and Kosecka, J. 2017. Synthesizing training data for object detection in indoor scenes. *arXiv*, preprint arXiv: 1702.07836.
- Girshick, R., Donahue, J., Darrell, T., and Malik, J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. *In 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 580–587. IEEE.
- Girshick, R. B., Felzenszwalb, P. F., and McAllester, D. 2012. Discriminatively Trained Deformable Part Models, Release 5.
- Grothues, T. M., Dobarro, J., Ladd, J., Higgs, A., Niezgodka, G., and Miller, D. 2008. Use of a multi-sensored AUV to telemeter tagged Atlantic sturgeon and map their spawning habitat in the Hudson River, USA. *In Autonomous Underwater Vehicles, 2008. AUV 2008. IEEE/OES*, pp. 1–7. IEEE.
- Gupta, A., Vedaldi, A., and Zisserman, A. 2016. Synthetic data for text localisation in natural images. *In IEEE Conference on Computer Vision and Pattern Recognition*.
- Kingma, D. P., and Ba, J. 2014. Adam: a method for stochastic optimization. *arXiv*, preprint arXiv: 1412.6980.
- Lau, P. Y., Correia, P. L., Fonseca, P., and Campos, A. 2008. I2N2: a software for the classification of benthic habitats characteristics. *In 2008 16th European Signal Processing Conference*, pp. 1–5. IEEE.
- Mahmood, A., Bennamoun, M., An, S., Sohel, F. A., Boussaid, F., Hovey, R., Kendrick, G. A., et al. 2018. Deep image representations for coral image classification. *IEEE Journal of Oceanic Engineering*, 44: 121–131.
- Ohn-Bar, E., and Trivedi, M. M. 2017. Multi-scale volumes for deep object detection and localization. *Pattern Recognition*, 61: 557–572.


- Pedersen, M., Bruslund Haurum, J., Gade, R., and Moeslund, T. B. 2019. Detection of marine animals in a new underwater dataset with varying visibility. *In* Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 18–26.
- Pérez, P., Gangnet, M., and Blake, A. 2003. Poisson image editing. *ACM Transactions on Graphics*, 22: 313–318.
- Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. 2016. You only look once: Unified, real-time object detection. *In* Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788.
- Redmon, J., and Farhadi, A. 2018. Yolov3: an incremental improvement. arXiv, preprint arXiv: 1804.02767.
- Tan, C. S., Lau, P. Y., Correia, P. L., and Campos, A. 2018. Automatic analysis of deep-water remotely operated vehicle footage for estimation of Norway lobster abundance. *Frontiers of Information Technology & Electronic Engineering*, 19: 1042–1055.
- Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., and Abbeel, P. 2017. Domain randomization for transferring deep neural networks from simulation to the real world. *In* 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 23–30. IEEE.
- Tolimieri, N., Clarke, M. E., Singh, H., and Goldfinger, C. 2008. Evaluating the SeaBed AUV for monitoring groundfish in untrawlable habitat. *In* Marine Habitat Mapping Technology for Alaska, pp. 129–141.
- Tremblay, J., Prakash, A., Acuna, D., Brophy, M., Jampani, V., Anil, C., To, T., *et al.* 2018. Training deep networks with synthetic data: bridging the reality gap by domain randomization. *In* Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 969–977.
- Wang, D., Vinson, R., Holmes, M., Seibel, G., and Tao, Y. 2018. Convolutional neural network guided blue crab knuckle detection for autonomous crab meat picking machine. *Optical Engineering*, 57: 043103.
- Wang, X., Yang, M., Zhu, S., and Lin, Y. 2013. Regionlets for generic object detection. *In* Proceedings of the IEEE International Conference on Computer Vision, pp. 17–24.
- Wei, X.-S., Xie, C.-W., Wu, J., and Shen, C. 2018. Mask-CNN: localizing parts and selecting descriptors for fine-grained bird species categorization. *Pattern Recognition*, 76: 704–714.
- Williams, S. B., Pizarro, O., How, M., Mercer, D., Powell, G., Marshall, J., and Hanlon, R. 2009. Surveying nocturnal cuttlefish camouflage behaviour using an AUV. *In* IEEE International Conference on Robotics and Automation, 2009. ICRA'09, pp. 214–219. IEEE.
- Williams, S. B., Pizarro, O., Jakuba, M. V., Mahon, I., Ling, S. D., and Johnson, C. R. 2010. Repeated AUV surveying of urchin barrens in North Eastern Tasmania. *In* 2010 IEEE International Conference on Robotics and Automation (ICRA), pp. 293–299. IEEE.

Handling editor: Cigdem Beyan

Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Original Article

Identifying the species of harvested tuna and billfish using deep convolutional neural networks

Yi-Chin Lu¹, Chen Tung¹, and Yan-Fu Kuo ^{1*}

¹Department of Bio-Industrial Mechatronics Engineering, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei, 106, Taiwan

*Corresponding author: tel: +886 2 3366 5329; fax: +886 2 2362 7620; e-mail: ykuo@ntu.edu.tw

Lu, Y.-C., Tung, C., and Kuo, Y.-F. Identifying the species of harvested tuna and billfish using deep convolutional neural networks. – ICES Journal of Marine Science, 77: 1318–1329.

Received 15 November 2018; revised 21 February 2019; accepted 26 April 2019; advance access publication 3 June 2019.

Fish catch species provide essential information for marine resource management. Some international organizations demand fishing vessels to report the species statistics of fish catch. Conventionally, the statistics are recorded manually by observers or fishermen. The accuracy of these statistics is, however, questionable due to the possibility of underreporting or misreporting. This paper proposes to automatically identify the species of common tuna and billfish using machine vision. The species include albacore (*Thunnus alalunga*), bigeye tuna (*Thunnus obesus*), yellowfin tuna (*Thunnus albacares*), blue marlin (*Makaira nigricans*), Indo-pacific sailfish (*Istiophorus platypterus*), and swordfish (*Xiphias gladius*). In this approach, the images of fish catch are acquired on the decks of fishing vessels. Deep convolutional neural network models are then developed to identify the species from the images. The proposed approach achieves an accuracy of at least 96.24%.

Keywords: convolutional neural network, deep learning, fish species identification, fishery management, model visualization, transfer learning.

Introduction

Fish is a major dietary protein source. In 2014, ~81.5 million MT of aquatic products were harvested from marine sources worldwide (FAO, 2016). Because of the high demand and advancement in fishing technology, fishing grounds in the world have been tapped rapidly in the past two decades. The Food and Agriculture Organization of the United Nations reported that 31.4% of the fish stocks are overfished (FAO, 2016), showing that the management of fishery resources is extremely urgent. Hence, international organizations have begun regulating fishing practices by demanding vessels to report fish catch statistics, such as fish species (Hosch and Blaha, 2017). The statistics are usually manually recorded by observers or fishermen, and thus, their accuracy is questionable because they can be misreported or underreported. Therefore, an automated approach for fish species identification is required. Combined with electronic monitoring systems (Monteagudo *et al.*, 2015), the approach may be used to identify species of fish catches in images or videos automatically. Thus, the labor for reporting the fish catch statistics can be reduced and the accuracy of the reports can be improved.

Image analysis approaches have been increasingly used to collect fish species information. These approaches, in contrast to conventional manual methods, have benefits of automation, efficiency, truthfulness, and accuracy. Previous studies have addressed the identification of sea fish types using image analysis. Rodrigues *et al.* (2010) developed a nearest-neighbour classifier for identifying fish of nine species using morphological and colour traits. Hu *et al.* (2012) developed a directed acyclic graph multi-class support vector machine classifier for distinguishing fish of six species using wavelet-based texture features as the inputs. Li and Hong (2014) developed a method using image processing and statistical analysis for recognizing fish of four species with colour, shape, and textural traits. Navarro *et al.* (2016) assessed 27 fish morphological traits and found three types of fish to differ considerably from each other. Huang *et al.* (2015) combined hierarchical tree with Gaussian mixture model to recognize 15 species of fish in underwater videos. Marini *et al.* (2018) estimated the abundance of the fish using an autonomous imaging device and genetic-programming-based classifier. Another project, Fish4Knowledge (Fisher *et al.*, 2016), developed tools for

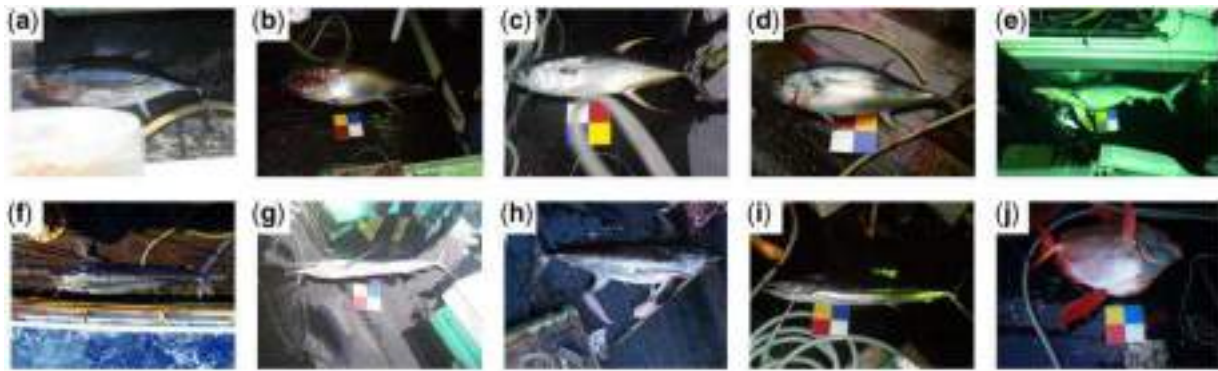


Figure 1. Images of (a) albacore, (b) big eye tuna, (c) yellowfin tuna, (d) other tuna, (e) shark, (f) blue marlin, (g) Indo-pacific sailfish, (h) swordfish, (i) other billfish, and (j) moonfish.

analysing the behaviours of fish in underwater videos using image processing and machine learning approaches. Although presumably accurate, these image analysis approaches typically use hand-crafted features (i.e. features defined manually). Preprocessing may be required if these methods are applied to images that are collected at locations with a high degree of variability in illumination conditions or of complex backgrounds.

Images of fish acquired on the deck of vessels are usually under uncontrolled conditions. Figure 1 shows fish images acquired on longliners: (i) albacore (ALB, *Thunnus alalunga*), (ii) bigeye tuna (BET, *Thunnus obesus*), (iii) yellowfin tuna (YFT, *Thunnus albacares*), (iv) southern bluefin tuna (*Thunnus maccoyii*), (v) blue shark (*Prionace glauca*), (vi) blue marlin (BUM, *Makaira nigricans*), (vii) Indo-pacific sailfish (SFA, *Istiophorus platypterus*), (viii) swordfish (SWO, *Xiphias gladius*), (ix) shortbill spearfish (*Tetrapturus angustirostris*), and (x) moonfish. The decks where the fish were located were full of miscellaneous items. Moreover, the illumination condition varies unavoidably because fishing is performed 24 h and weather is uncontrollable. Hence, it is challenging to use the aforementioned image analysis approaches for identifying the fish species from the images.

Recently, deep learning has emerged as a powerful tool for addressing complicated image analysis problems. Convolutional neural networks (CNNs; Fukushima, 1980) are a deep learning approach specifically used for image classification. CNNs are multilayer perceptron composed of millions of neurons. The neurons are arranged as sets of filters to perform spatial convolution. After training the parameters of the neurons, the convolution operations can extract desired features from the input images with almost no preprocessing. Hence, CNNs are used to tackle complex classification problems. Initially, CNNs were used to perform tasks on images with a simple background, such as handwritten character recognition (Bengio et al., 1994), mammogram masses and normal tissue distinction (Wei et al., 1995), textural pattern classification (Tivive et al., 2006), and face recognition (Lawrence et al., 1997). With the advances in graphic processing unit (GPU) computing, CNNs became larger and deeper and have been applied to solving complicated tasks. Krizhevsky et al. (2012) developed a deep CNN for distinguishing images of 22 000 classes in 2012 ILSVRC. Lee et al. (2017) developed a CNN-based system for identifying 1000 species of plants in the 2016 plantCLEF task. Sprengel et al. (2016) developed a deep CNN model for recognizing 999 species of birds from monophonic

recordings in the 2016 BirdCLEF challenge. Although presumably powerful, thousands of images are normally required for training deep CNNs, which may restrict the use of deep CNNs.

Transfer learning has alleviated the demand for a large amount of training data for CNNs (Pan and Yang, 2010). Originally, transfer learning aimed to transfer knowledge between related sources and target domains (Caruana, 1995). Starting from this concept, it has been shown that models trained using huge datasets can be adopted for other applications because the first layers of neural networks deal with generic features (Yosinski et al., 2014). Oquab et al. (2014) exhibited the high potential of using the mid-level features extracted from networks trained using the ImageNet dataset for classifying images in the Pascal VOC 2007 and 2012 datasets. Li et al. (2015) detected fish and recognized the species of the fish in the images of the ImageCLEF dataset using pre-trained CNNs and fast region-based CNN. Siddiqui et al. (2017) identified 16 species of fish in underwater videos using pre-trained CNNs. Ali-Gombe et al. (2017) recognized fish species in images with random noise using CNNs and transfer learning.

This study aimed to automatically identify the species of major tuna and billfish from the images acquired on longliners. The specific objectives were to (i) collect images of major tuna and billfish fish, (ii) adapt pre-trained deep CNN models for identifying the fish species, (iii) demonstrate the performance of the models, and (iv) visualize the features learned by the CNN models.

Material and methods

Image collection

A total of 16 517 images of fish catch were provided by Fishery Agency, Council of Agriculture (Taiwan). The images were acquired on the deck of longliners by observers between 2006 and 2017 using digital cameras. The illumination conditions when the images were taken varied considerably. Some images were acquired during dark nights using flash light (Figure 1b), while others were acquired on sunny days (Figure 1f). Shadows may cover part of the fish body (Figure 1a). The images were sorted into ten categories: ALB, BET, YFT, other tuna (OT), BUM, SWO, SFA, other billfish (OB), shark, and other fish (OF) (Table 1). The category of OT contained two species: southern bluefin tuna and Skipjack tuna (*Katsuwonus pelamis*). The category of OB contained four species: striped marlin fish

(*Kajikia audax*), giant black marlin (*Makaira indica*), shortbill spearfish, and longbill spearfish (*T. pfluegeri*). The category of OT contained common sea fish other than tuna, billfish, or shark (e.g. dolphin fish, moonfish, and smooth skin oilfish).

Image preprocessing, cross-validation, and image augmentation

The dimensions of the fish images ranged from 640×360 to 4608×3456 pixels. To reduce the complexity of the CNN models, the images were resized to 330×250 pixels. Zero padding was

applied to the resized images for maintaining the aspect ratio of the images. Subsequently, image augmentation was applied to the images for model training (i.e. training images). Image manipulation generalizes the images and, hence, increases the robustness of the models to be developed. The augmentation operations included horizontal flipping, vertical flipping, width shifting (randomly between -33 and 33 pixels), height shift (randomly between -25 and 25 pixels), rotation (randomly between 0° and 30°), shearing (randomly between 0 and 66 pixels), zoom-in (randomly between 1 and 1.2), and zoom-out (randomly between 0.8 and 1) (Figure 2). Each operation was randomly applied to the images before they were used for training.

Table 1. Numbers of images for each fish species or type.

Species/type	Numbers of images
Albacore (ALB, <i>Thunnus alalunga</i>)	2 240
Big eye tuna (BET, <i>Thunnus obesus</i>)	2 240
Yellowfin tuna (YFT, <i>Thunnus albacares</i>)	2 240
Other tuna (OT)	1 735
Blue marlin (BUM, <i>Makaira nigricans</i>)	1 056
Indo-pacific sailfish (SFA, <i>Istiophorus platypterus</i>)	416
Swordfish (SWO, <i>Xiphias gladius</i>)	1 600
Other billfish (OB)	830
Shark	1 600
Other species of fish (OF)	2 560

Strategies for fish species identification

Two strategies were used for fish species identification. Strategy one used three models in a cascade (Figure 3). Model 1A was used to identify fish types: tuna, billfish, shark, and OF. Models 1B and 1C, respectively, were used to identify the species of tuna and billfish. Strategy two used a single model (Model 2) to identify fish types and fish species for tuna and billfish. Strategy one alleviated the issue of unbalanced image numbers (Table 1) in model training.

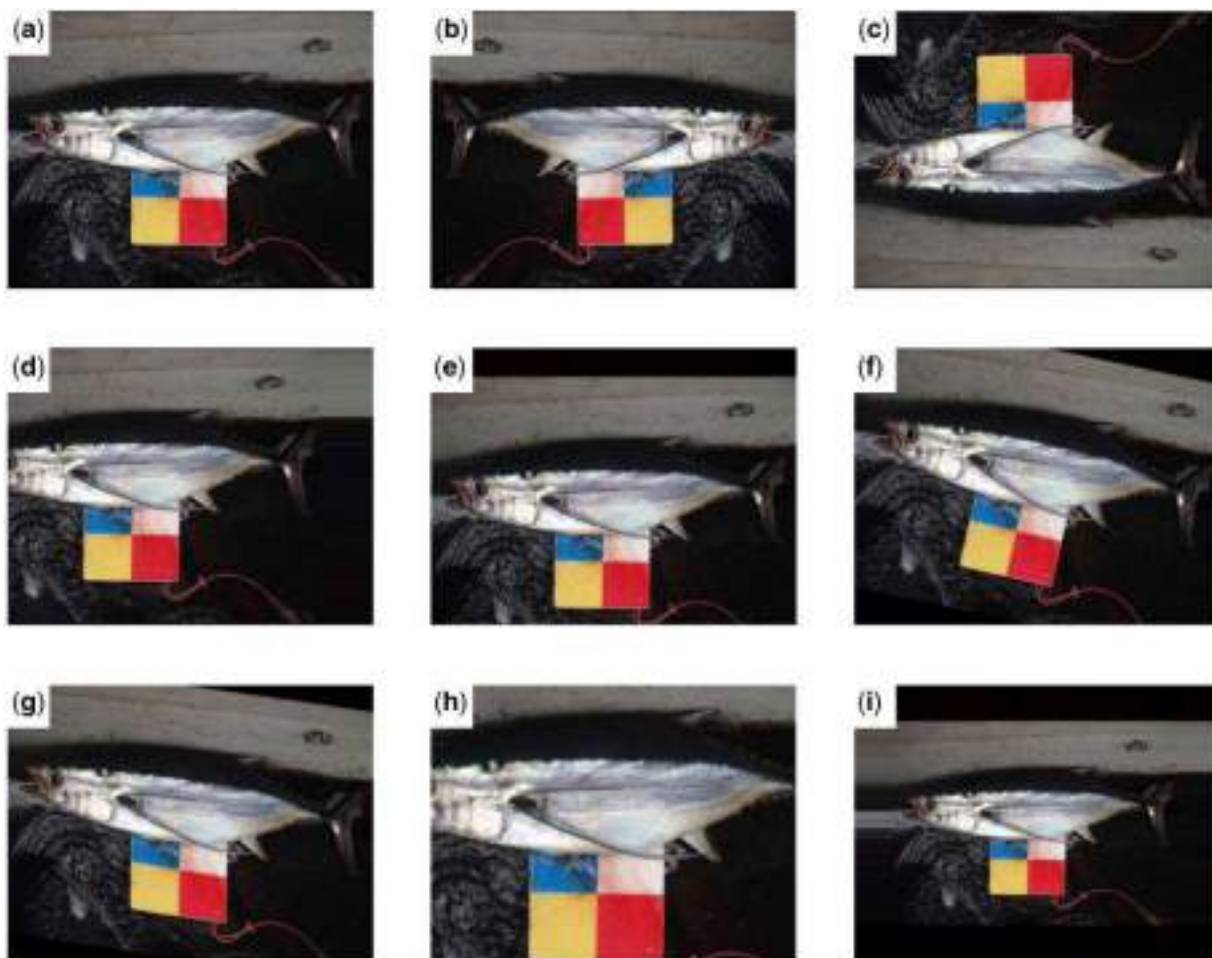


Figure 2. Image manipulation: (a) original image, (b) horizontal flipping, (c) vertical flipping, (d) width shift, (e) height shift, (f) rotation, (g) shearing, (h) zoom-in, and (i) zoom-out.

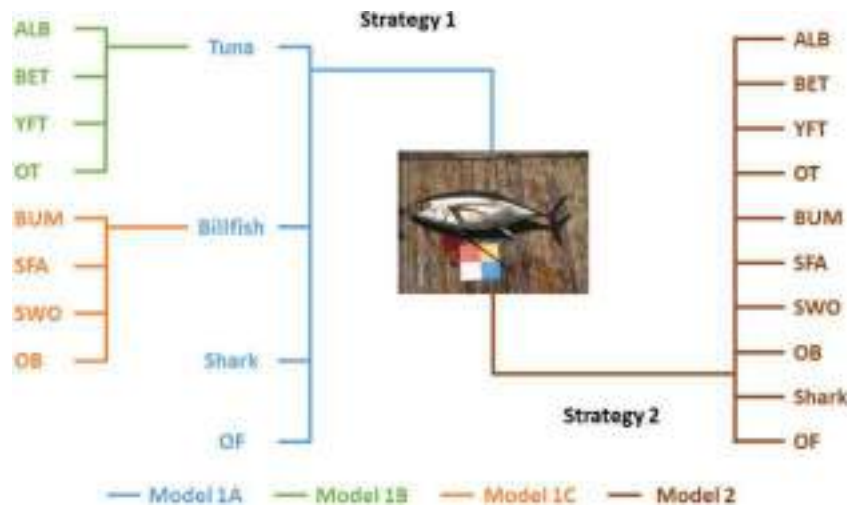


Figure 3. Two strategies for fish type and species identification. Strategy 1 uses three models to identify fish types, tuna species, and billfish species. Strategy 2 uses a single model to identify fish types and species.

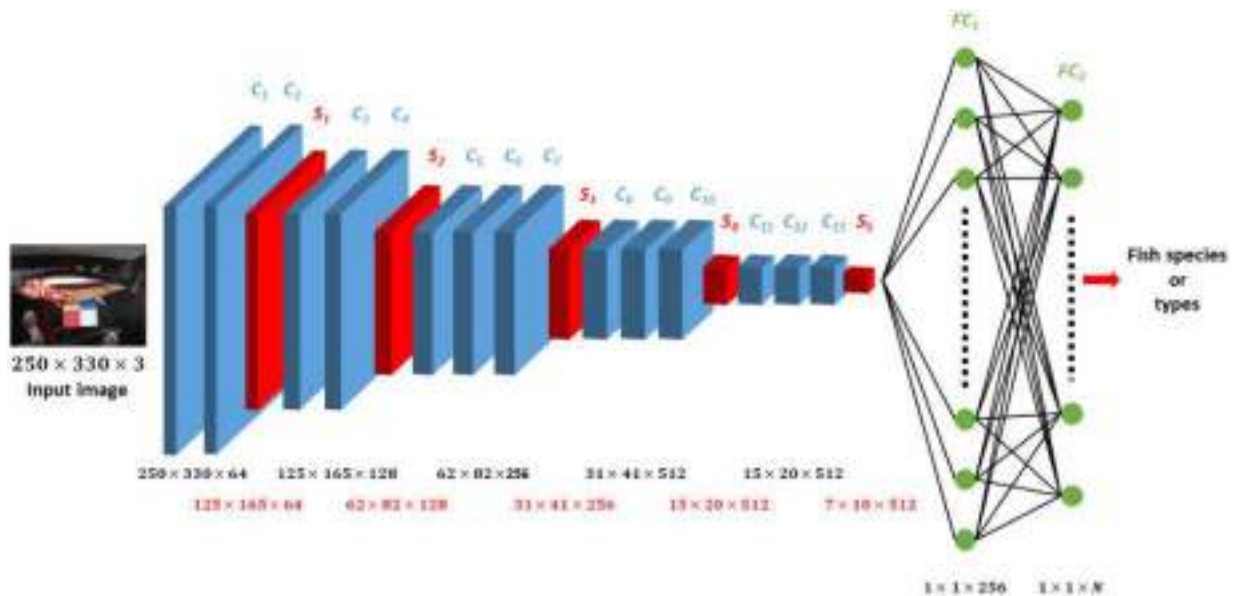


Figure 4. Architecture of the modified VGG-16 model. C: convolution layer, S: max pooling layer, and FC: fully connected layer.

Model development using transfer learning

Transfer learning was applied to the development of deep CNN models. In this procedure, a model with parameters pre-trained using other datasets was adapted. The structures of the output layers were modified to match the output dimensions (i.e. types or species). Next, some layers of the model were frozen. Fine-tuning was then applied to the remaining layers of the model to update the parameters. In this study, VGG-16 (Simonyan and Zisserman, 2014) was chosen as the pre-trained model because the architecture performed well in various classification tasks and was used in numerous applications (Ballas et al., 2015; Liu et al., 2016; Lopez et al., 2017; Abas et al., 2018). Originally, VGG-16 consisted of 13 convolutional (C₁ to C₁₃), 5 max pooling (S₁ to S₅), and 3 fully connected (FC₁ to FC₃) layers (Figure 4). A convolutional layer applies convolution operations to the neurons in

the current layer using filters and passes the results to the next layer. A pooling layer combines the neurons in the current layer into a single neuron in the next layer (Huang et al., 2007). A fully connected layer connects every neuron in the current layer to every neuron in the next layer (Viglione, 1970). Convolutional layers C₁ to C₁₃ contained 64, 64, 128, 128, 256, 256, 256, 512, 512, 512, 512, 512, and 512 filters, respectively. The dimension and stride of the filters in the convolutional layers were 3 x 3 pixels and 1 pixel, respectively. Zero padding was used in the convolution operations to keep the dimension of the output the same as that of the input. The dimension and stride of the filters in the max pooling layers were 2 x 2 pixels and 2 pixels, respectively.

In this study, the architecture of VGG-16 was adjusted by replacing the original FC layers with new FC layers (FC₁ and FC₂ in Figure 4) with dimension of R²⁵⁶ and R^N, where N is the

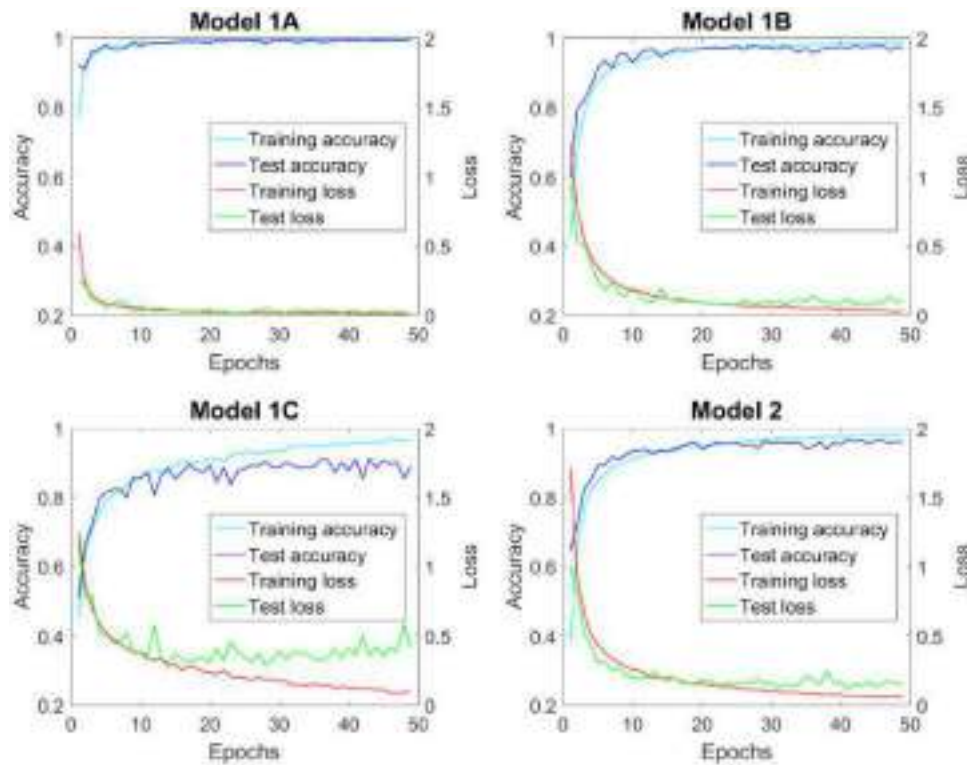


Figure 5. Model accuracy and loss during training.

number of categories to be classified in each model. The rectified linear unit (ReLU; [Glorot et al., 2011](#)) was used as the activation function for all convolutional layers and the first FC layer. Softmax ([Bishop, 1995](#)) was used as the activation function for the second FC layer to determine the confidence scores of the predicted fish types or species. In this study, parameters in the first four convolutional layers (C_1 to C_4) were frozen, while those in the remaining layers (C_5 to FC_2) were fine-tuned during training.

Model training

The models were developed using adaptive moment estimation ([Kingma and Ba, 2014](#)) as the optimizer and cross-entropy as the loss function. The initial learning rate was set to 0.00002. Each model was trained for 50 epochs. In each epoch, image augmentation was randomly applied to the training images. Effectively, the images were augmented for 50 times. The models were then trained using the images and back propagation ([Rumelhart et al., 1986](#)). To prevent the models from being overfitted, dropout ([Srivastava et al., 2014](#)) with a rate of 0.5 was applied to layer FC1. Hence, in the training stage, each neuron in FC1 had 50% chance of being ignored. The model development was performed using Python3 and Keras toolbox ([Chollet, 2015](#)). A GPU (GeForce GTX 1080 Ti, NVIDIA; Santa Clara, USA) was used to expedite the training. Tenfold cross-validation ([Kohavi, 1995](#)) was applied for assessing the performance of the models. The mean accuracies were presented.

Visualization of filters in the CNN models

Filters of the CNN model were visualized to realize how the CNN models work and what features the models had learned. To visualize a specific filter in a CNN model, a loss function that

maximizes the activation of the filter was determined. An image with a dimension of 330×250 pixels was next generated and initialized with random pixel values. The gradient of the loss function using the image as the input to the CNN model was calculated. Gradient ascent ([Simonyan et al., 2013](#)) was then applied to update the pixel values in the input image. The aforementioned steps were performed for 200 iterations. The resulting input image was the visualization of the filter.

Saliency maps and Grad-CAMs of the CNN models

Saliency maps ([Simonyan et al., 2013](#)) and gradient-weighted class activation maps (Grad-CAMs; [Selvaraju et al., 2017](#)) were generated to illustrate the essential information in an input image for the developed models to determine the category (i.e. fish types or species) of the image. Saliency maps indicate the importance of each pixel in an input image. In the procedure of calculating a saliency map, an input image of a known category was fed into a trained CNN model. The derivatives of the model output with respect to the pixels of the input image were calculated using guided backpropagation ([Springenberg et al., 2014](#)). The saliency map was then formed as the derivatives reshaped to the dimension of the input image (i.e. 330×250). Grad-CAM indicates the importance of pixels in the feature maps of a model. In the procedure of calculating a Grad-CAM, an input image of a known category was fed into a developed CNN model. The gradients of the model output with respect to the feature maps of the last convolutional layer in the model were calculated, and then, the gradients were fed into global average pooling ([Lin et al., 2013](#)). The weighted combination of the feature maps using the gradients as the weights were calculated.

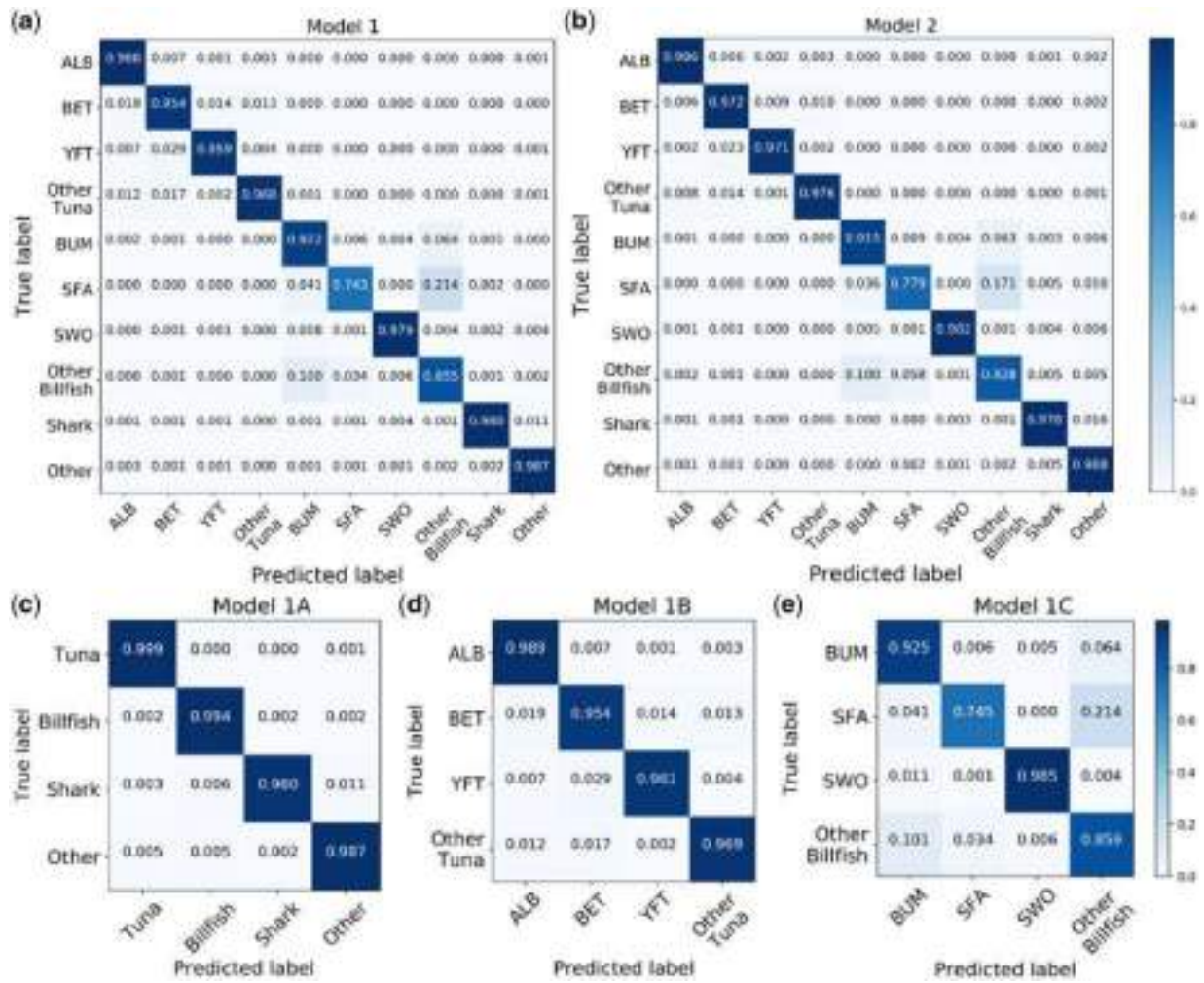


Figure 6. Test accuracy of (a) Model 1, (b) Model 2, (c) Model 1A, (d) Model 1B, and (e) Model 1C.

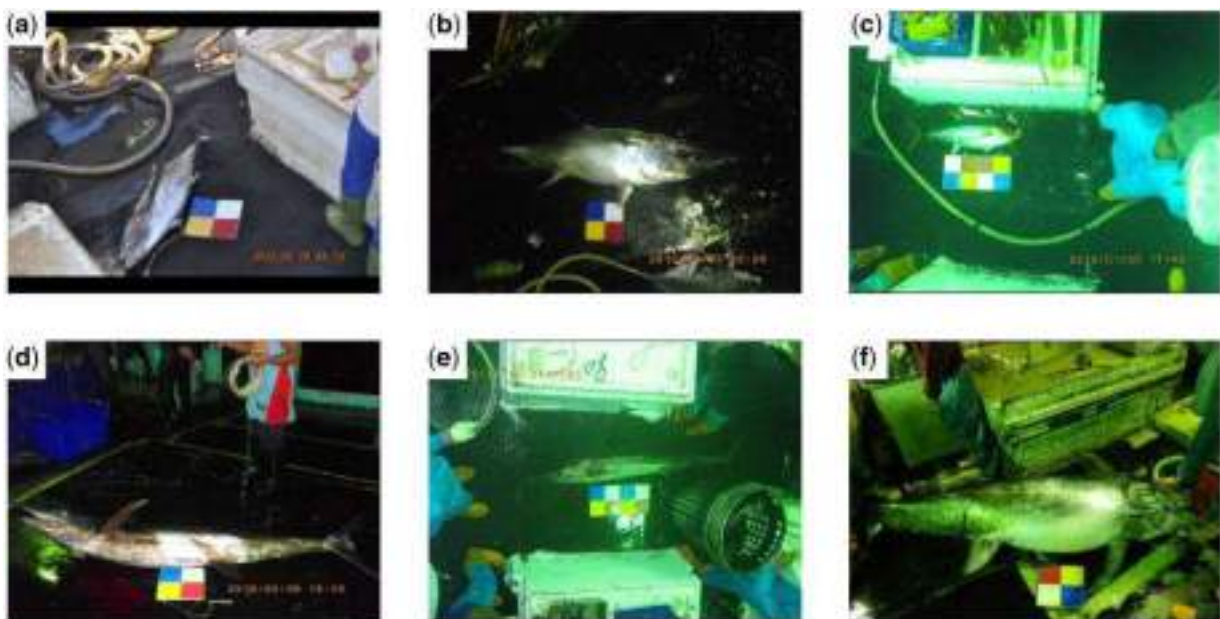


Figure 7. Challenging cases that were successfully identified: (a) ALB, (b) BET, (c) YFT, (d) BUM, (e) SFA, and (f) SWO.

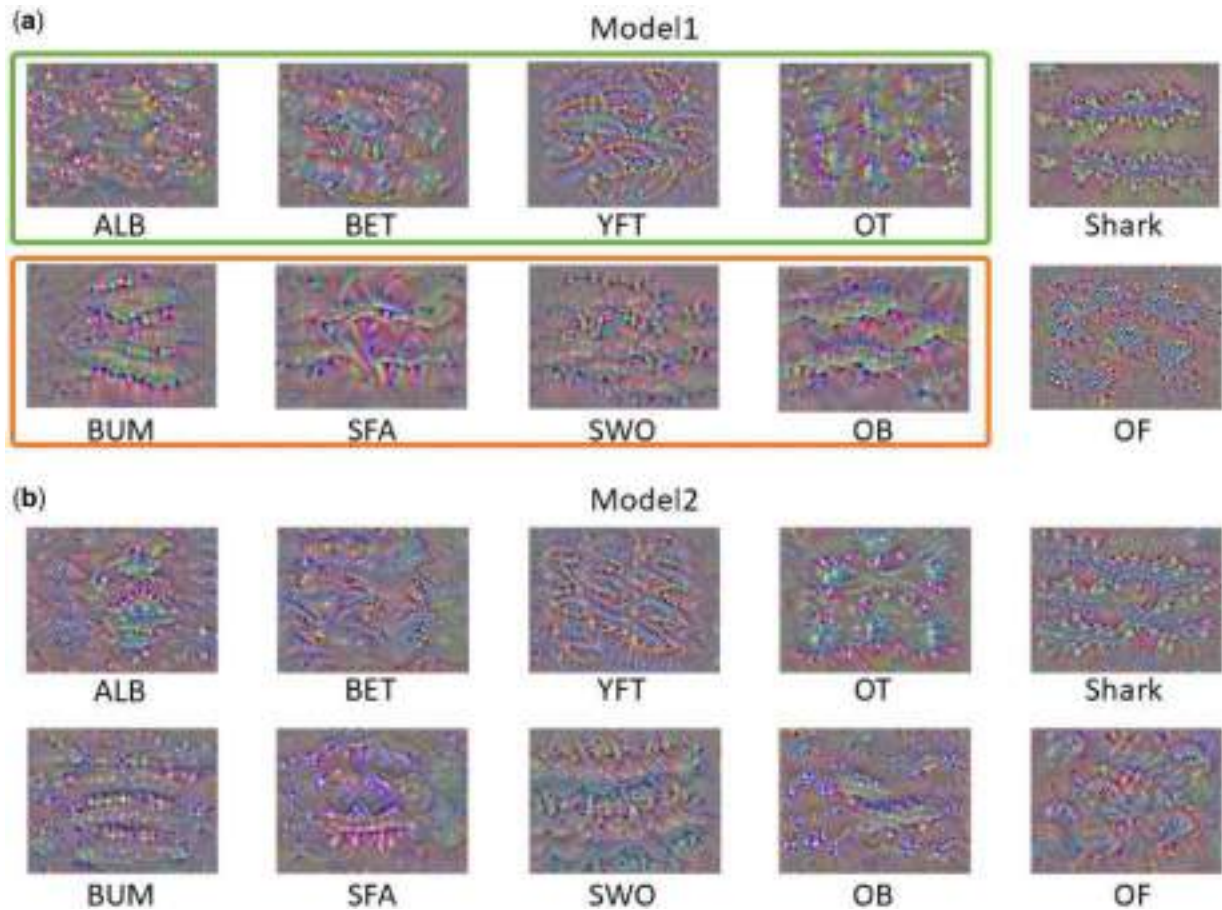


Figure 8. Visualization of the last fully connected filters of each species or type. The green and orange boxes enclose the visualization of filters in Models 1B and 1C, respectively.

Grad-CAM was the output of the ReLU function using the weighted combination as the input.

Fish species identification using bag-of-features approach

A bag-of-features (BoF; Sivic and Zisserman, 2003) model was developed as the baseline for performance comparison with the proposed CNN-based approach. In the BoF model, the size of the visual vocabulary was set to 1000. Speeded-up robust features (Bay et al., 2008) with a Hessian threshold of 1000 were used as the features. Soft-margin support vector machines (SVMs, Chang and Lin, 2011) with radial basis function kernels were used as the classifiers. The SVMs were arranged in the one-vs.-rest fashion to fulfill the task of multiclass classification. The margin and kernel parameters of the SVMs were determined using grid search.

Results and discussion

Model accuracy and loss during training

The accuracies and losses of the models during training were examined (Figure 5). After 50 epochs, both the training and test losses of Models 1A, 1B, and 2 converged to under 0.16. Both the training and test accuracies of Models 1A, 1B, and 2 reached over 96%. However, for Model 1C, there was ~6% difference between the training and test accuracies. This observation implied that

Model 1C might be slightly overfitted, which could be caused by the inadequate amount of training images (Table 1). The issue of overfitting may be resolved by increasing the amount of the training images of SFA.

Performance of the models

The performance of the developed CNN models was evaluated using tenfold cross validation (Figure 6). In the evaluation, Models 1A, 1B, and 1C were concatenated to form Model 1 (Figure 6a). The mean accuracies of Models 1 and 2 were 95.85% and 96.24%, respectively. The standard deviations of the accuracies were 0.75% and 0.67% for Models 1 and 2, respectively. The mean processing time for Models 1 and 2 to classify an image were 0.0226 s and 0.0155 s, respectively, using a GPU (GeForce GTX 1080 Ti). Models 1 and 2 used 8575 MB and 8063 MB, respectively, of the GPU memory. Model 2 achieved higher accuracy and used less resource. However, Model 1 could provide the correct fish type of an image even if the fish species was misclassified. For both models, the two least accurate categories were SFA and OB (Figure 6a and b). The low accuracies in these two categories were also observed in Model 1C (Figure 6e), which may be caused by the imbalanced training images (i.e. only 416 images for SFA and 830 images for OB; Table 1).

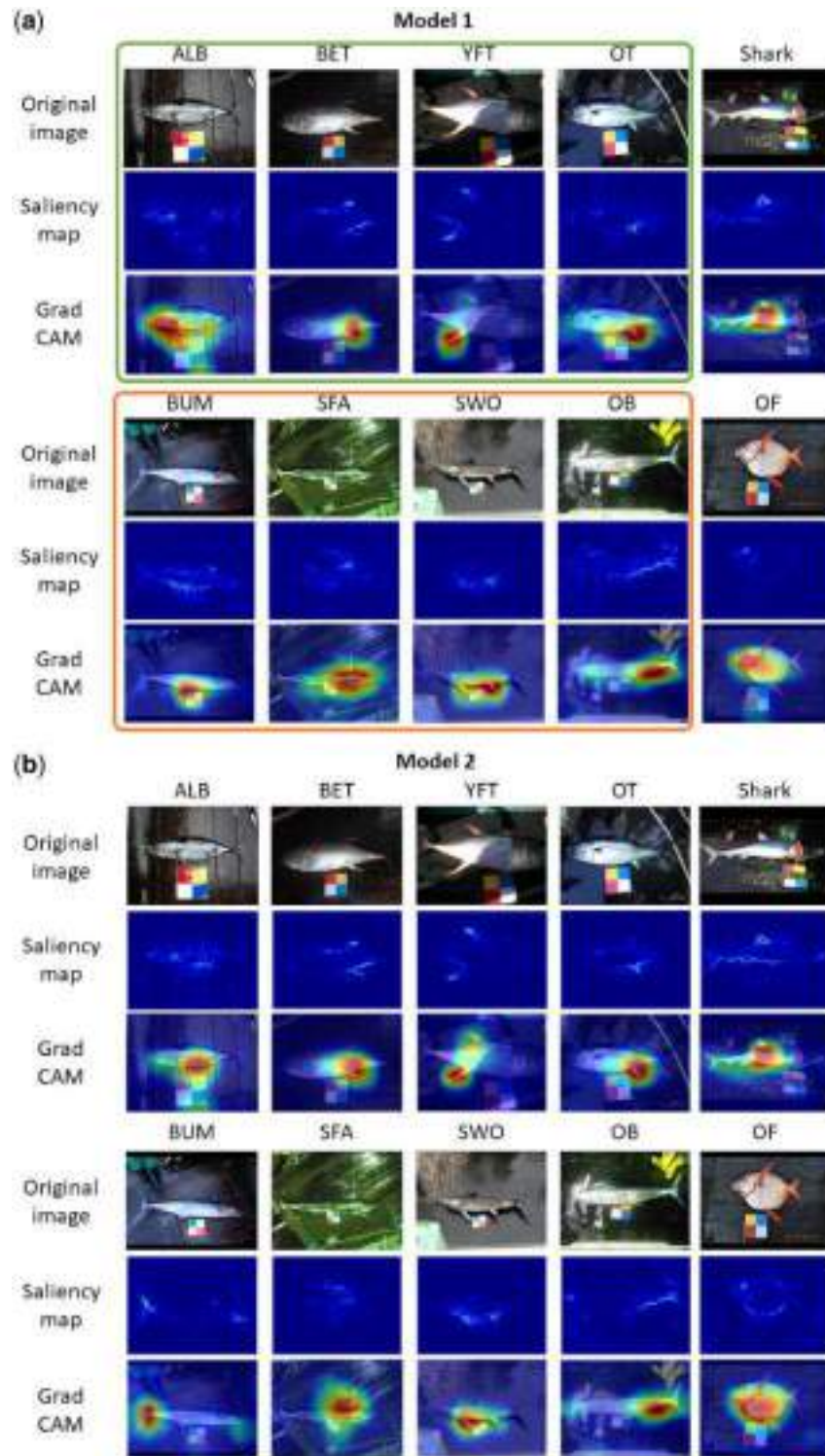


Figure 9. Saliency maps and Grad-CAM of (a) Model 1 and (b) Model 2. The green and orange boxes enclose the visualization of filters in Models 1B and 1C, respectively.

Cases that were challenging to be identified were examined. Figure 7 illustrates the images of ALB, BET, YFT, BUM, SFA, and SWO that were successfully identified. The challenges included panned fish body (Figure 7a), low lamination (Figure 7b

and d), colour tone shifting (Figure 7c, e, and f), inadequate resolution (Figure 7c), slanted fish body (Figure 7d), and incomplete fish body (Figure 7f). In Figure 7, the upper jaw of SWO was cut off.

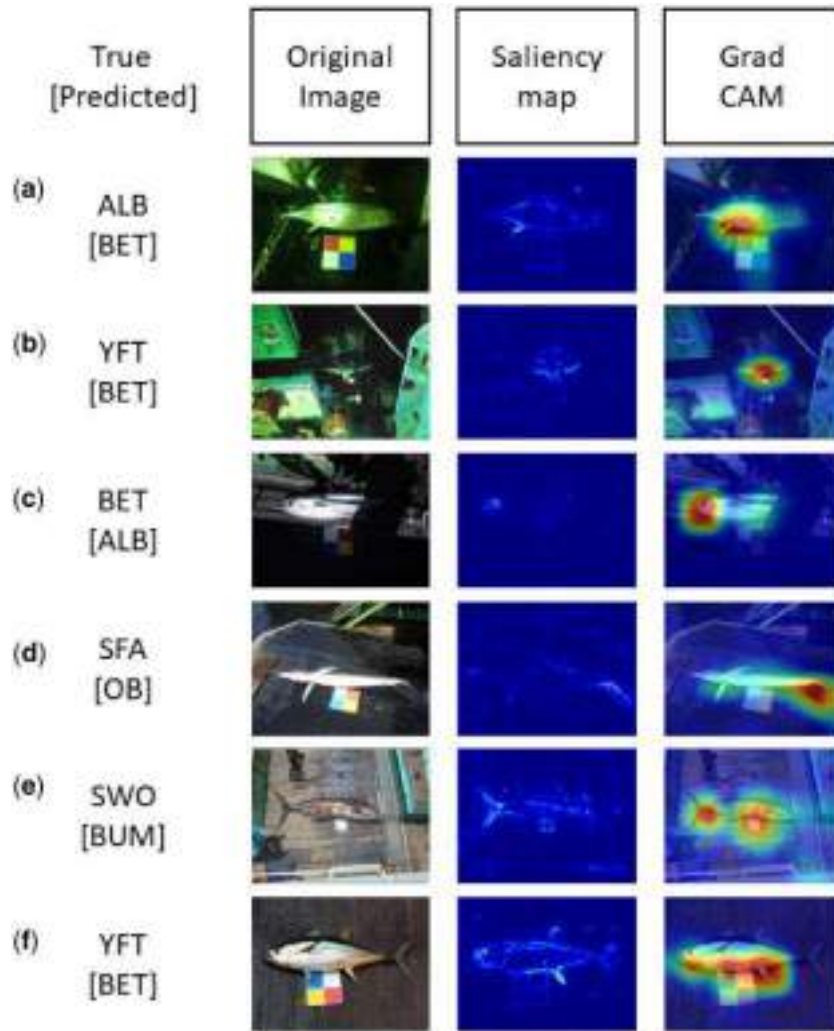


Figure 10. Misclassified cases. The true and predicted categories of the images were shown on the left side.

Filters of the CNN models

Filters of the last FC layer in Models 1 and 2 were visualized (Figure 8). The filters in both models exhibited patterns similar to parts of the fish body of each fish species or type. The filters of tuna (ALB, BET, YFT, and OT) displayed curves and sawtooth waves corresponding to the dorsal and anal fins and finlets, respectively, of tuna. The filters of billfish (BUM, SFA, SWO, and OB) displayed patterns similar to the dorsal fin and anal fins and long upper jaw. The filters of shark exhibited patterns corresponding to the first dorsal fin of shark. The filters of OF displayed patterns of fish body contours, which were distinct from those of tuna, billfish, or shark.

The pattern differences between the tuna species were observed. Yellow curves similar to dorsal fins of tuna appeared in the filters of YFT and BET; however, they were not found in the filters of ALB and OT. In addition, the curves in YFT filters were much longer than those in BET filters. Moreover, the horizontal strips in OT filters were similar to the grain patterns on the bodies of Skipjack tuna. The same patterns were not found in ALB, BET, and YFT filters. The aforementioned characteristics may be the benchmarks for the models to distinguish the tuna species.

The pattern differences between the billfish species were also observed. The patterns of body contours were found in the filters of BUM, SWO, and OB, but not in those of SFA. In addition, the dorsal fin patterns were observed in the filters of all billfish categories; however, SWO filters exhibited the most substantial patterns of dorsal fins compared with BUM, SFA, and OB filters. Moreover, the dorsal fin patterns were displayed in SFA filters, but not in BUM, SWO, and OB filters.

Saliency maps and Grad-CAMs of the CNN models

The saliency maps and Grad-CAMs of the developed models were generated (Figure 9). The same set of fish images was used as input to the two models for comparison purposes. The saliency maps displayed that the models paid attention mostly to the contour, pectoral fin, finlets, dorsal fins, and anal fins of the fish, while Grad-CAMs displayed that the models paid attention mostly to the abdomen, dorsum, and anal fins of the fish.

For the tuna species, the ALB maps displayed that the pectoral fins received considerable attention. This observation agreed with the fact that ALB has longer pectoral fins compared with the remaining tuna species (Chapman *et al.*, 2015). The OT maps showed that only anal fins received attention. By contrast, the

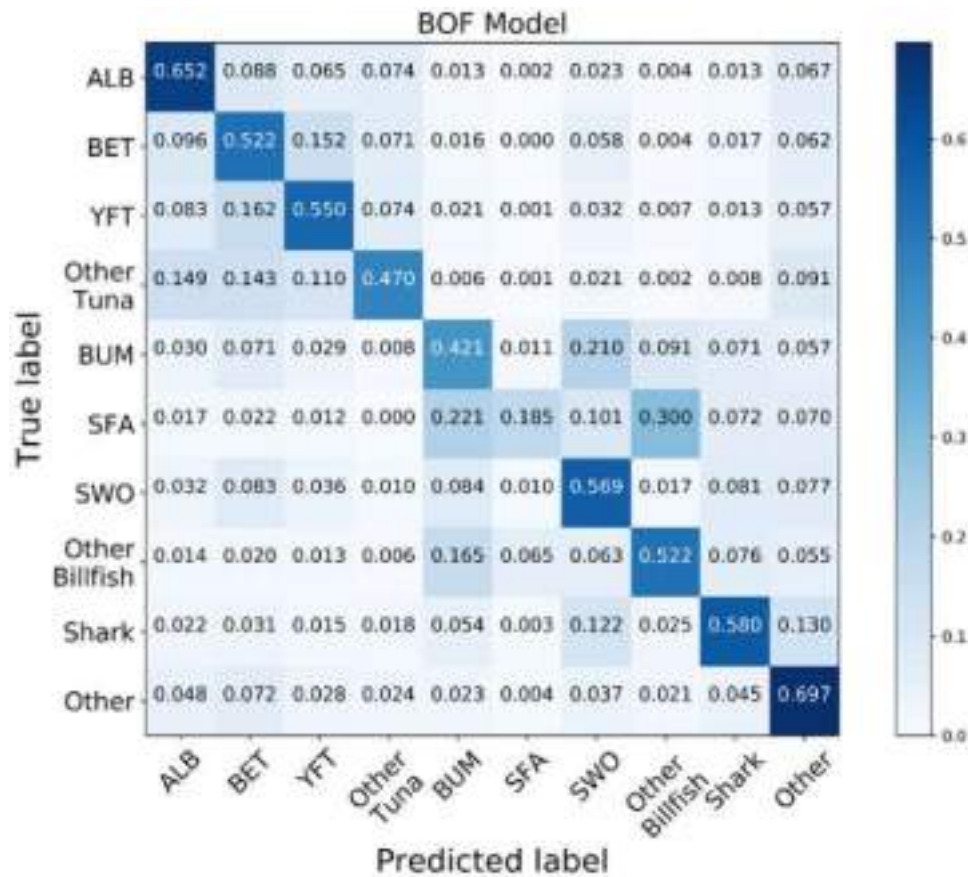


Figure 11. Test accuracy of the BoF model.

maps of YFT and BET displayed that both the dorsal and anal fins received considerable attention. Particularly, the attention to the dorsal and anal fins of YFT was strong. This observation agreed with the fact that YFT has longer second dorsal and anal fins compared with BET. Moreover, the BET maps showed that the finlets received considerable attention. This observation agreed with the fact that BET finlets are bright yellow with a black edge. The areas that received strong attention agreed with the characteristics of human observers for distinguishing the tuna species.

For the billfish species, the SWO maps displayed that the pectoral fins and first anal fins received considerable attention. This observation agreed with the fact that the pectoral fins of SWO can flatten against its body, whereas those of BUM and SFA cannot. The maps of SFA displayed that the first dorsal fins received considerable attention. This observation agreed with the fact that SFA has a large first dorsal fin. The width of its first dorsal fin can be double its body width (Chapman *et al.*, 2015). The BUM maps displayed that the abdomen, tail and head received considerable attention. BUM has two caudal keels, whereas SWO has only one. In addition, the dorsal fin of BUM is not as large as that of SFA. These differences were used to distinguish BUM from SWO and SFA.

For shark and OF, the first dorsal fins and body contours received considerable attention. The dorsal fins of shark are usually larger than those of tuna, billfish and OF. Moreover, the contours of shark fins are smooth, whereas those of tuna, billfish and OF

fins are tippy. This information was used to distinguish shark and OF from tuna or billfish.

Study of misclassification cases

Misclassification occurred due to colour tone variation, inadequate resolution, low illumination, body part occlusion, or fish immaturity. Figure 10a displays an image of ALB that was falsely recognized as BET. The image was acquired at night and was in green tone. The pectoral fin, one of the most essential traits of ALB, of the fish were almost invisible. The saliency map and Grad-CAM of the image confirmed that the pectoral fin received almost no attention. Instead, the anal fin received attention. Figure 10b displays an image of YFT that was falsely recognized as BET. The image was in green tone and was taken from a distance. The saliency map and Grad-CAM of the image indicated that the fish contour was not completely identified. The ventral of the fish received attention at a certain degree. However, the dorsal and anal fins, two of the most essential traits of YFT, received almost no attention. Figure 10c displays an image of BET that was falsely recognized as ALB. Shadow covered the tail of the fish body and made the finlets invisible. The saliency map and Grad-CAM of the image displayed that the fish contour was not completely identified. Although the anterior of the fish received attention at a certain degree, the part typically does not contain traits that are essential for determining the species. Figure 10d displays an image of SFA that was falsely recognized as OB. The

body of the fish was tilted so that the dorsal fin, one of the most essential traits of SFA, of the fish was occluded. The saliency map and Grad-CAM of the image displayed that the posterior received attention. However, the posterior of the fish typically does not contain traits that are essential for determining the species. **Figure 10e** displays an image of SWO that was falsely recognized as BUM. The colour of the second anal fin of the fish was similar to that of the background, making the second anal fin almost invisible. Also, the pectoral fin was close to the fish body, making it almost invisible. The saliency map and Grad-CAM of the image confirmed that the second anal fin or pectoral fin of the fish did not receive strong attention. **Figure 10f** displays an image of YFT at juvenile stage. The saliency map and Grad-CAM of the image displayed that the contour of the fish was clearly identified and the dorsal and anal fin of the fish received strong attention. However, the lengths of the fins were short. Thus, YFT was falsely recognized as BET. Although misclassified, a tuna species was usually falsely recognized as another tuna species and a billfish species was usually falsely recognized as another billfish species (**Figure 10**).

The performance of the bag-of-features model

The performance of the BoF model was evaluated using tenfold cross validation (**Figure 11**). The mean accuracy reached 56.03% and the standard deviation of the accuracy was 1.69%. The majority of the misclassification cases occurred within the same fish types. A tuna species was usually falsely recognized as another tuna species, and a billfish species was usually falsely recognized as another billfish species. This observation indicated that the BoF model could distinguish fish with obvious differences in appearance, such as fish type. However, the model could not effectively recognize the subtle differences in appearance between the fish species of the same type.

Conclusions

This paper proposed the identification of the species of six common tuna and billfish using machine vision. In the proposed approach, images of fish catch were acquired on the deck of longliners with miscellaneous items in the background and under various illumination conditions. The images were then resized to 330×250 pixels with zero padding. CNN models were next developed to identify the fish species using a pre-trained architecture VGG-16 and the concept of transfer learning. Saliency maps and Grad-CAMs of the models exhibited that the information the models learned were the characteristics that human observers used for distinguishing the fish species. The proposed approach outperformed conventional BoF approaches and reached an overall accuracy of at least 96.24%.

Funding

This research was supported by the Fish Agency, Council of Agriculture, Taiwan, under the grants 106AS-18.1.7-FA-F1 and 107AS-14.2.7-FA-F1.

References

Abas, M. A. H., Ismail, N., Yassin, A. I. M., and Taib, M. N. 2018. VGG16 for plant image classification with transfer learning and data augmentation. *International Journal of Engineering and Technology (UAE)*, 7: 90–94.

Ali-Gombe, A., Elyan, E., and Jayne, C. 2017, August. Fish classification in context of noisy images. *In International Conference on*

Engineering Applications of Neural Networks, pp. 216–226. Springer, Cham.

Ballas, N., Yao, L., Pal, C., and Courville, A. 2015. Delving deeper into convolutional networks for learning video representations. *arXiv preprint arXiv:1511.06432v4*.

Bay, H., Ess, A., Tuytelaars, T., and Van Gool, L. 2008. Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110: 346–359.

Bengio, Y., LeCun, Y., and Henderson, D. 1994. Globally trained handwritten word recognizer using spatial representation, convolutional neural networks, and hidden Markov models. *Advances in Neural Information Processing Systems*, 6: 937.

Bishop, C. M. 1995. *Neural Networks for Pattern Recognition*. Oxford University Press, New York, NY.

Caruana, R. 1995. Learning Many Related Tasks at the Same Time with Backpropagation, pp. 657–664. MIT Press, Cambridge, MA.

Chang, C. C., and Lin, C. J. 2011. LIBSVM: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2: 27.

Chapman, L., Sharples, P., Brogan, D., Desurmont, A., Beverly, S., and Sokimi, W. 2015. *Marine species identification manual for horizontal longline fishermen Taiwanese-English*.

Chollet, F. and others. 2015. keras. <https://keras.io> (last accessed March 2018).

FAO. 2016. *The State of World Fisheries and Aquaculture 2016. Contributing to Food Security and Nutrition for All*. Rome. 200 pp.

Fisher, R., Chen-Burger, Y.-H., Giordano, D., Hardman, L., and Lin, F.-P. 2016. *Fish4Knowledge: Collecting and Analyzing Massive Coral Reef Fish Video Data*. (Intelligent Systems Reference Library; Vol. 104). Springer International Publishing, New York, NY.

Fukushima, K. 1980. Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36: 193–202.

Glorot, X., Bordes, A., and Bengio, Y. 2011. Deep sparse rectifier neural networks. *In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pp. 315–323.

Hosch, G., and Blaha, F. 2017. *Seafood traceability for fisheries compliance – countrylevel support for catch documentation schemes*. FAO Fisheries and Aquaculture Technical Paper No. 619. Rome, Italy.

Hu, J., Li, D., Duan, Q., Han, Y., Chen, G., and Si, X. 2012. Fish species classification by color, texture and multi-class support vector machine using computer vision. *Computers and Electronics in Agriculture*, 88: 133–140.

Huang, F. J., Boureau, Y. L., and LeCun, Y. 2007. Unsupervised learning of invariant feature hierarchies with applications to object recognition. *In Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pp. 1–8. IEEE.

Huang, P. X., Boom, B. J., and Fisher, R. B. 2015. Hierarchical classification with reject option for live fish recognition. *Machine Vision and Applications*, 26: 89–102.

Kingma, D. P., and Ba, J. 2014. Adam: a method for stochastic optimization. *arXiv preprint arXiv:1412.6980v9*.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *In Advances in Neural Information Processing Systems*, pp. 1097–1105.

Kohavi, R. 1995. A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, Vol. 2, Montreal, pp. 1137–1145.

Lawrence, S., Giles, C. L., Tsoi, A. C., and Back, A. D. 1997. Face recognition: a convolutional neural-network approach. *IEEE Transactions on Neural Networks*, 8: 98–113.

- Lee, S. H., Chang, Y. L., and Chan, C. S. 2017. Lifeclef 2017 plant identification challenge: classifying plants using generic-organ correlation features. Working Notes of CLEF, 2017.
- Li, L., and Hong, J. 2014. Identification of fish species based on image processing and statistical analysis research. *In* Mechatronics and Automation (ICMA), 2014 IEEE International Conference on, pp. 1155–1160. IEEE.
- Li, X., Shang, M., Qin, H., and Chen, L. 2015. Fast accurate fish detection and recognition of underwater images with fast R-CNN. *In* OCEANS'15 MTS/IEEE Washington, pp. 1–5. IEEE.
- Lin, M., Chen, Q., and Yan, S. 2013. Network in network. arXiv preprint arXiv:1312.4400v3.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., and Berg, A. C. 2016, October. Ssd: single shot multibox detector. *In* European Conference on Computer Vision, pp. 21–37. Springer, Cham.
- Lopez, A. R., Giro-i-Nieto, X., Burdick, J., and Marques, O. 2017, February. Skin lesion classification from dermoscopic images using deep learning techniques. *In* Biomedical Engineering (BioMed), 2017 13th IASTED International Conference on, pp. 49–54. IEEE.
- Marini, S., Corgnati, L., Mantovani, C., Bastianini, M., Ottaviani, E., Fanelli, E., Aguzzi, J. *et al.* 2018. Automated estimate of fish abundance through the autonomous imaging device GUARD1. *Measurement*, 126: 72–75.
- Monteagudo, J. P., Legorburu, G., Justel-Rubio, A., and Restrepo, V. 2015. Preliminary study about the suitability of an electronic monitoring system to record scientific and other information from the tropical tuna purse seine fishery. *Collective Volumes of Scientific Papers ICCAT*, 71: 440–459.
- Navarro, A., Lee-Montero, I., Santana, D., Henríquez, P., Ferrer, M. A., Morales, A., Soula, M. *et al.* 2016. IMAFISH_ML: a fully-automated image analysis software for assessing fish morphometric traits on gilthead seabream (*Sparus aurata* L.), meagre (*Argyrosomus regius*) and red porgy (*Pagrus pagrus*). *Computers and Electronics in Agriculture*, 121: 66–73.
- Oquab, M., Bottou, L., Laptev, I., and Sivic, J. 2014. Learning and transferring mid-level image representations using convolutional neural networks. *In* Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1717–1724.
- Pan, S. J., and Yang, Q. 2010. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22: 1345–1359.
- Rodrigues, M. T., Padua, F. L., Gomes, R. M., and Soares, G. E. 2010, September. Automatic fish species classification based on robust feature extraction techniques and artificial immune systems. *In* Bio-Inspired Computing: Theories and Applications (BIC-TA), 2010 IEEE Fifth International Conference on, pp. 1518–1525. IEEE.
- Rumelhart, D. E., Hinton, G. E., and Williams, R. J. 1986. Learning representations by back-propagating errors. *Nature*, 323: 533.
- Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. 2017. Grad-CAM: visual explanations from deep networks via gradient-based localization. 2017 IEEE International Conference on Computer Vision (ICCV), Venice, 2017, pp. 618–626.
- Siddiqui, S. A., Salman, A., Malik, M. I., Shafait, F., Mian, A., Shortis, M. R., and Harvey, E. S. 2017. Automatic fish species classification in underwater videos: exploiting pre-trained deep neural network models to compensate for limited labelled data. *ICES Journal of Marine Science*, 75: 374–389.
- Simonyan, K., Vedaldi, A., and Zisserman, A. 2013. Deep inside convolutional networks: visualising image classification models and saliency maps. arXiv preprint arXiv:1312.6034v2.
- Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556v6.
- Sivic, J., and Zisserman, A. 2003. Video Google: a text retrieval approach to object matching in videos. *In* null, p. 1470. IEEE.
- Sprengel, E., Jaggi, M., Kilcher, Y., and Hofmann, T. 2016. Audio based bird species identification using deep learning techniques. *In* LifeCLEF 2016 (No. EPFL-CONF-229232, pp. 547–559).
- Springenberg, J. T., Dosovitskiy, A., Brox, T., and Riedmiller, M. 2014. Striving for simplicity: the all convolutional net. arXiv preprint arXiv:1412.6806v3.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15, 1929–1958.
- Tivive, F. H. C., and Bouzerdoum, A. 2006. Texture classification using convolutional neural networks. *In* TENCON 2006. 2006 IEEE Region 10 Conference, pp. 1–4. IEEE.
- Viglione, S. S. 1970. 4 Applications of pattern recognition technology. *In* Mathematics in Science and Engineering, 66, pp. 115–162. Elsevier.
- Wei, D., Sahiner, B., Chan, H. P., and Petrick, N. 1995. Detection of masses on mammograms using a convolution neural network. *In* Acoustics, Speech, and Signal Processing, 1995. ICASSP-95. 1995 International Conference on, 5, pp. 3483–3486. IEEE.
- Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. 2014. How transferable are features in deep neural networks?. *In* Advances in Neural Information Processing Systems, pp. 3320–3328.


Handling editor: Cigdem Beyan



Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Original Article

Image-based, unsupervised estimation of fish size from commercial landings using deep learning

Amaya Álvarez-Ellacuría^{1*}, Miquel Palmer¹, Ignacio A. Catalán¹, and Jose-Luis Lisani ²

¹IMEDEA (CSIC-UIB), Illes Balears, Spain

²Universitat de les Illes Balears, Illes Balears, Spain

*Corresponding author: tel: +34 971611978; e-mail: amaya@imedea.uib-csic.es.

Álvarez-Ellacuría, A., Palmer, M., Catalán, I. A., and Lisani, J.-L. Image-based, unsupervised estimation of fish size from commercial landings using deep learning. – ICES Journal of Marine Science, 77: 1330–1339.

Received 6 May 2019; revised 18 October 2019; accepted 22 October 2019; advance access publication 23 November 2019.

The dynamics of fish length distribution is a key input for understanding the fish population dynamics and taking informed management decisions on exploited stocks. Nevertheless, in most fisheries, the length of landed fish is still made by hand. As a result, length estimation is precise at fish level, but due to the inherent high costs of manual sampling, the sample size tends to be small. Accordingly, the precision of population-level estimates is often suboptimal and prone to bias when properly stratified sampling programmes are not affordable. Recent applications of artificial intelligence to fisheries science are opening a promising opportunity for the massive sampling of fish catches. Here, we present the results obtained using a deep convolutional network (Mask R-CNN) for unsupervised (i.e. fully automatic) European hake length estimation from images of fish boxes automatically collected at the auction centre. The estimated mean of fish lengths at the box level is accurate; for average lengths ranging 20–40 cm, the root-mean-square deviation was 1.9 cm, and maximum deviation between the estimated and the measured mean body length was 4.0 cm. We discuss the challenges and opportunities that arise with the use of this technology to improve data acquisition in fisheries.

Keywords: convolutional neural networks, deep learning, fish size estimation, landings

Introduction

Ensuring fish stocks sustainability while maximizing fishers profitability is an elusive and still not solved topic (Hilborn, 2007; Iudicello *et al.*, 2012). Solving this puzzle is specially urgent in the case of the Mediterranean fleets because they have been going through a deep crisis for decades, which has been attributed to the continuous decrease in the sale price of fish, that translates in a continuous decrease in the number of boats and in a very low recruitment rate of young fishermen (Palmer *et al.*, 2017). At least in the case of the Balearic Islands, the root of the problem seems to be more related to the commercialization of the product than to the state of conservation of the stocks (Reglero and Morales-Nin, 2008; Morales-Nin *et al.*, 2010; Maynou *et al.*, 2013).

Comanagement is one of the strategies aimed to match stocks sustainability and fisher profits (d'Armengol *et al.*, 2018). Several small-scale fisheries are currently comanaged in the Balearic Islands (e.g. *Aphia minuta* and *Coryphaena hippurus*) suggesting that fishers are prone to adopt this type of strategy. Comanagement delivers both ecological and social benefits (d'Armengol *et al.*, 2018), but periodically updated reviews of the results are mandatory in order to adopt short-term, operational management decisions. Moreover, those decisions must be informed by accurate and precise data. Similarly, conventional fishery models may inform on the mid- and long-term trends of the exploited stocks but must be fed with accurate and precise data too.

Fish length is one of the key variables needed for both taking short-term management decisions and modelling stock trends. Nevertheless, in almost all fisheries, the estimation of the length of landed fish is still made by hand. Length measures are precise enough for those purposes, but since the observers cost is high, the sample size used for estimating length at population level tends to be relatively small. Accordingly, the estimation of the length distribution at any given time may be imprecise and may be prone to bias when properly stratified sampling programmes are not affordable. In contrast with the relatively low efficiency of observers, a massive amount of images can be processed by computer vision.

Hake (*Merluccius merluccius*) is considered overfished in the Mediterranean, with an alarming 20% reduction in catches in the last 20 years. Overfishing in the Balearic islands has been considered moderate (FAO, 2016), but the overall status in the Mediterranean is considered critical (FAO, 2018). Moreover, this species represents an economically relevant fraction of the landings in the Balearic Islands (Palmer *et al.*, 2009, 2017). Accordingly, in this study, we propose the hake as case study species for implementing computer vision techniques for massively estimating fish length from images. In particular, we propose to adapt an existing convolutional neural network (CNN, Mask R-CNN; He *et al.*, 2017) to the problem at hand. This strategy is technically feasible in Mallorca (Balearic Islands) because images of fish boxes are routinely obtained at the conveyor belt, just at the bidding, in the auction centre. Therefore, at any port with similar facilities (fish on a conveyor belt is common practice elsewhere), length estimates for all the fish boxes sold in a day, all the days of the year, could be obtained at affordable cost, thus fully fulfilling the data requirements that would enable taking informed operational decisions at the short-term scale needed for comanagement and, at the same time, monitoring the mid- and long-term trends of the stocks.

Some of the earliest attempts at using computer vision techniques for length measurement of fish were reported by Arnarson *et al.* (1991) and Strachan (1993). In both cases, a camera was placed on top of a conveyor belt where fishes passed by, one at a time. The illumination conditions were controlled in such a way that fishes were much darker than the background. Therefore, a simple illumination threshold was used to detect fish. Once detected, their orientation was determined and normalized and the (possibly curved) line from nose to tail fork was computed. The length of this line was used to estimate the actual length of the fish. More complex versions included edge detection, color calibration and the distinction between roundfish and flatfish (White *et al.*, 2006). However, the system setting (conveyor belt with controlled lighting) remained similar. In a similar way, in Abdullah *et al.* (2009) pictures of individual fishes were used as input and edge-and-corner detection methods were applied to estimate the position of head and tail from which the length was computed.

Detection and measurement of live fish in underwater images is more challenging. Concerning detection, body silhouettes have been extracted using edge detection techniques under controlled illumination conditions (Hardin, 2006; Zion *et al.*, 2007; Miranda and Romero, 2017). Stereo methods and 3D models have been proposed to concurrently estimate the fish length and the distance of the free-swimming fish from the camera (Petrell *et al.*, 1997; Tillet *et al.*, 2000; Díaz-Gil *et al.*, 2017). Image

enhancement techniques for the correction of color and illumination have also been implemented (Martinez-de Dios *et al.*, 2003; Costa *et al.*, 2006; Al-Jubouri *et al.*, 2017). A common characteristic of these methods is that, once the distance to the camera and the illumination have been normalized, they use classical image processing techniques (segmentation by thresholding or edge/corner detection) to extract the fish's features of interest.

However, those conventional image processing techniques have been progressively replaced by methods based on machine learning for the tasks of detection and classification. One of the first applications of machine learning was the detection of human faces (Viola and Jones, 2004). Subsequently, techniques based on learning algorithms called support vectors machines, and the use of local image descriptors (Dalal and Triggs, 2005) obtained notable results in the detection of various types of objects (faces, vehicles, pedestrians, etc.). The use of CNNs for pattern recognition received a strong boost in 1990 with the use of new optimization techniques for network training (Lecun *et al.*, 1998).

In 2012, the use of graphical processing units (GPUs) allowed the implementation of CNNs with many layers (deep networks) and trained on large amounts of data that exceeded human performance in image classification tasks (Krizhevsky *et al.*, 2012), giving rise to the current boom of deep learning (GPUs are hardware devices that speed-up the computations needed to train a neural network).

Since then, increasingly deep networks have been proposed and have been applied to detection, classification and segmentation. Some of the most popular deep learning models for detection are YOLO (Redmon *et al.*, 2016) and Mask R-CNN (He *et al.*, 2017).

In the case of fish detection, the use of deep learning techniques is incipient and faces the additional problem that fish are not rigid objects and networks must learn how to adapt to changes in posture, position and scale. Nevertheless, fish recognition has been achieved using a binary classifier (Marini *et al.*, 2018) or a neural network with only two convolutional layers (Qin *et al.*, 2016). In French *et al.* (2015, 2019), a CNN was designed for counting fish in video. In addition, existing network architectures (e.g. LeNet, AlexNet, GoogLeNet, and YOLO) have been used for fish classification (Chen *et al.*, 2017; Meng *et al.*, 2018; Villon *et al.*, 2018). In Monkman *et al.* (2019), the authors describe a system the measurement of fish detected using R-CNNs.

The goal of the present study is to automatically obtain the fish length from fish box images obtained in the ports. In our case study, hakes are arranged inside a fish box in such a way that in most cases the tails are occluded and a complete view is available for a few fish only (see Figure 2). Accordingly, in the case of hake, the target object to be detected cannot be the whole fish but only a part. Fortunately, many complete heads are visible on the images; thus, fish heads have been the target object with which the network has been trained.

In our implementation, we use a similar network architecture than recently published papers (French *et al.*, 2019; Monkman *et al.*, 2019), the main difference being the set of pictures used for training. This set of pictures must necessarily be different for each application, since the network must be fine-tuned for each specific task. Another important difference is the final goal of each system. In our case, we want to measure the fish, even when they are partially occluded. In (Monkman *et al.*, 2019) they seek a

similar goal, but they use pictures from online sources for training (not pictures from the auction centre, as we do), their CNN does not provide a segmentation of the images and, more importantly, the developed system cannot cope with occlusions. In French *et al.* (2019), the goal is to classify fish in video from CCTV cameras installed on fishing trawlers. Since their goal is identification and not measurement, they can use either partial detections or full detections, as long as the detected parts permit to identify the fish. Their system does not need to deal with the problem of inferring the whole length of the fish from the partial detections. Contrarily to previous published works on the subject of fish length estimation, the system proposed in the current paper is able to deal with partial occlusions and develops different statistical models that permit to estimate the total fish length from the length of the detected heads.

Material and methods

Images

Three sets of images of hake boxes were used for the study. The photos were obtained with the same webcam (pixel resolution of 1280×760). The first set (562 images) of hake boxes was obtained at the conveyor belt in the auction centre of Palma. The camera was placed top-down, just over the fish boxes and the images were taken at the bidding moment, when the conveyor belt stops for a while. The second set (56 images) was obtained at the laboratory with the same camera setting. For the network implementation, 163 randomly selected images from the first set and 14 randomly selected images from the second set were used. Of these total 177 images of both sets used in the network, up to 2112 heads for the training steps and 490 heads for the validation steps of the network (a total of 2602 heads) were manually annotated using the LABELBOX software (<https://labelbox.com/>). The head has been defined here as the area from the mouth to the pelvic fin (Figure 1).

The detection performance of the trained network was assessed using 42 images from the second set, containing 200 visible heads that had not been used for training. These 200 fish were also used to implement a statistical model relating the total fish length (in centimetres) to the head length (in pixels) measured from the output of the network.

Finally, the third set (10 images) was also obtained at the conveyor belt in the auction centre. These images were neither used for training the network nor for building the statistical model. The model-based estimates of fish length (centimetres) obtained for the fish heads detected by the network on these images were compared with the actually measured fish lengths

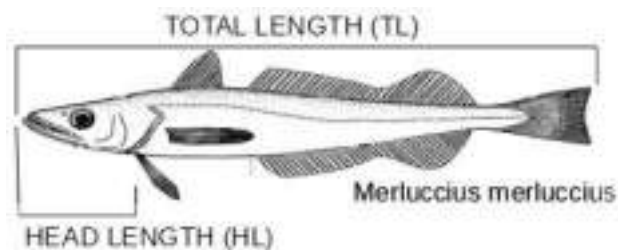


Figure 1. Head definition: from the mouth to the pelvic fin (modified from European Commission data; https://mare.istc.cnr.it/fisheriesv2/javax.faces.resource/images/species/HKE_l.jpg.xhtml).

(centimetres) for assessing the accuracy and precision of the whole system.

Figure 4 gives an overview of the sampling protocol used to train the neural network, build the statistical model and measure the performance of the proposed system.

Provided that fishermen sort hake and most of the landings in boxes by species, there is no need for any preliminary classification task. Some examples of the images used as input of the system are displayed in Figure 2.



Figure 2. Examples of input images. Note that whereas the head is visible from a ventral side, the body tends to be partly hidden.

Network implementation

The network used (Mask R-CNN; He et al., 2017) is a simple, flexible, and general purpose network for object instance segmentation, which implies not only recognizing all the objects (or instances) from the target category in a given image but also accurately segmenting them. Mask R-CNN is based on Faster R-CNN (Ren et al., 2015), which focuses in object detection (i.e. each target object is enclosed into a rectangular bounding box), with an extension for creating a segmentation mask of the target object within the bounding box.

Mask R-CNN consists of two CNNs that work in parallel: a “backbone architecture” for the extraction of features over the entire image and a “head architecture” for recognizing regions of interest and producing a mask over them. A scheme of the network architecture is displayed in Figure 3. The developers of Mask R-CNN have demonstrated that the proposed architecture outperforms more complex networks and that the best results are

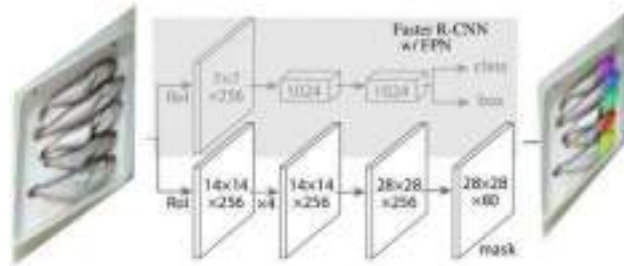


Figure 3. MASK R-CNN Architecture [modified from He et al. (2017)].

obtained with a ResNet-FPN (Lin et al., 2017) backbone with 101 layers and a fully convolutional network head consisting of six convolutional layers.

The implementation of Mask R-CNN used is available at Github (https://github.com/matterport/Mask_RCNN). Moreover, the network uses pre-trained weights from the COCO dataset (a public dataset of images available at <http://cocodataset.org/>). However, these weights must be fine-tuned for each case-specific target using a user-defined dataset.

The network has been fine-tuned using 2602 heads (Figure 4). Concerning the training process, the setting was 100 epochs, with 200 steps per epoch, and 50 validation steps. The learning rate was 0.002 and only weights of the head branch of the network were learned using the training set.

Evaluation metrics

The performance of the system is assessed in two ways. First, the *detection performance* of the network is evaluated in terms of the percentage of false positives among all the detected objects. A detection is deemed a false positive if the detected object is not a fish head; conversely, a false negative is a fish head that goes undetected. Note that in the specific context of our research, false positives are more relevant than false negatives since we have at our disposal a huge amount of data (a large number of boxes may be photographed each day, each box containing several fish) and, even if we miss some correct fish heads, provided that most of the detections are correct we shall be able to build an accurate statistical model of the fish length distribution.

Secondly, the *measurement performance* of the system is assessed by comparing the obtained length estimations with the actual values of length, after manual measurement of the fish.

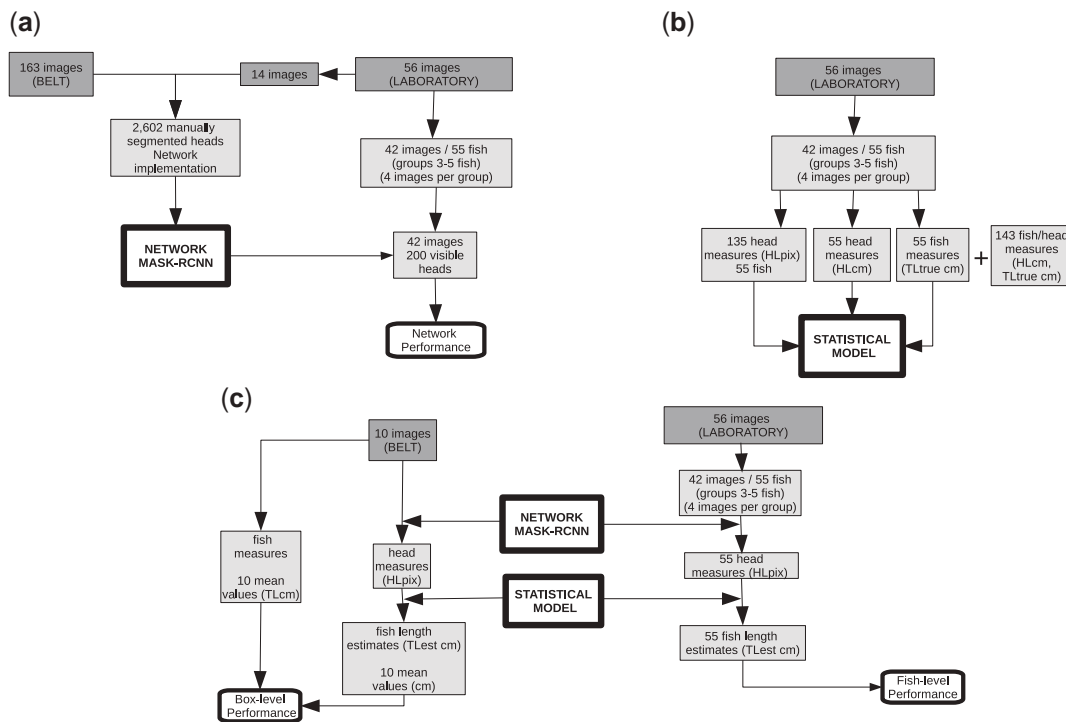


Figure 4. General overview of sampling design and analyses workflow. The figures in the top row describe the set of samples used for training and validation of the neural network (a) and for computing the parameters of the statistical model (b). The figure at the bottom (c) depicts the process for computing the system performance both at fish level and at box level (see text for details).

The system computes the length of a fish as follows: (i) the length (in pixels) of the detected head is computed (let us denote this value HL_{pix}), (ii) the length (in centimetres) of the detected head is computed from HL_{pix} using a statistical model (let us denote this value HL_{cm}), and (iii) the total length (TL, in centimetres) of the fish is inferred from HL_{cm} using a second statistical model.

The sampling protocol and the statistical models have been designed to deal with three sources of variability, namely (i) variability between the repeated estimates of HL_{pix} from the same fish (i.e. repeated measures of HL_{pix} for the same fish from different images after changing fish posture), (ii) variability related to the relationship between HL_{pix} and HL_{cm} , and (iii) variability related to the relationship between HL_{cm} and TL.

The statistical model is structured into three submodels. The first submodel assumes that the j repeated measurement for HL_{pix} of the same fish i from different images after changing fish posture is normally distributed around a mean value $\overline{HL_{pix}_i}$, with a standard deviation $\sigma_{HL_{pix}}$ (hereafter, overlined quantities refer to the expected value):

$$HL_{pix_{i,j}} \sim N(\overline{HL_{pix}_i}, \sigma_{HL_{pix}}). \quad (1)$$

To estimate the parameters of this submodel 135 measures of HL_{pix} corresponding to 55 fish ($i=1-55$) were obtained. Specifically, fish were visually labeled with an ID and were placed on fish boxes in groups of three to five fish. Several images from each group were taken after changing the posture of all fish in the box. Those images were submitted to the unsupervised routine for head detection described in the previous section. The median number of repeated measures of HL_{pix} per fish was 3.

Concerning the relationship between HL_{pix} and HL_{cm} (second submodel), a linear model with zero intercept was considered:

$$\begin{aligned} \overline{HL_{cm}_i} &= \beta_{head} \overline{HL_{pix}_i} \\ HL_{cm_i} &\sim N(\overline{HL_{cm}_i}, \sigma_{HL_{cm}}) \end{aligned} \quad (2)$$

where β_{head} is the slope of the linear relationship. To estimate the parameters of the submodel, HL_{cm} values of the same 55 fish were measured with a ruler by the same observer. In a preliminary analysis, four repeated measures from 70 fish showed that the standard deviation of the observer's measurement error of either HL_{cm} or TL was 0.2 cm and that this measurement error was independent of the fish size. Accordingly, this uncertainty source was considered negligible and hereafter ignored.

Concerning the relationship between HL_{cm} and TL (third submodel), four linear models resulting from log-transforming or not these variables were compared. The model finally selected (see results in the next section) was as follows:

$$\begin{aligned} \log \overline{TL}_i &= \alpha_{body} + \beta_{body} \log(\overline{HL_{cm}_i}) \\ \log TL_i &\sim N(\log \overline{TL}_i, \sigma_{TL}) \end{aligned} \quad (3)$$

where α_{body} and β_{body} are the intercept and the slope of the linear relationship, respectively. To estimate the parameters of the submodel, TL values of the same 55 fish were measured with a ruler by the same observer (Figure 4). However, since that uncertainty at this level was larger than expected (see Figure 6), TL and HL_{cm} were measured for 143 additional fish. Therefore, the sample size for the third submodel was 198 fish.

The parameters of the integrated model (i.e. combining the three submodels into a single analysis) were estimated using a Bayesian approach and Markov chain Monte Carlo (MCMC) methods (Kruschke, 2010). Three independent chains were run. The convergence of the MCMC chains was assessed by visual inspection of the chains and was evaluated using the Gelman–Rubin statistic (Plummer et al., 2006). Virtually flat priors were used: normal distribution with zero mean and a huge variance was assumed for $\overline{HL_{pix}}$, β_{head} , α_{body} and β_{body} . Gamma distributions (rate = 0.01, scale = 0.01) were assumed for the tolerances of the three standard deviations ($\sigma_{HL_{pix}}$, $\sigma_{HL_{cm}}$ and σ_{TL}). The posterior distribution was estimated from at least 30 000 valid iterations after appropriate burning (the first 10 000 iterations were not included) and thinning (only one of the ten iterations were kept because at this thinning level MCMC did not show autocorrelation). Additional technical details are available at the R script provided in the Supplementary material, which, along with the input data, allow reproducing the results reported here.

The accuracy of the TL predictions obtained from new HL_{pix} measures was assessed in two ways. First, randomly selected measures of HL_{pix} for each one of the 55 fish available for submodels 1 and 2 were used to predict TL after properly propagating uncertainty at the three considered levels. The predicted value of TL was then compared with the actually measured TL by the observer (fish-level performance in Figure 4). Second, ten new images of hake boxes were obtained at the auction centre and were submitted to the unsupervised routine for fish head segmentation described in the previous section. Moreover, a random sample of the fish in each box was measured (TL, centimetres) by an observer. Provided that the fish for which HL_{pix} was available may be different from the fish for which TL was available, accuracy of the mean fish size at the box level was assessed instead of fish-level accuracy (box-level performance in Figure 4).

Results

The Mask R-CNN was successfully implemented according with the developers specifications and fine-tuned with a data set composed of 2602 manually segmented heads.

Concerning the detection performance of the implemented system, in the 42 photos used as input a total of 200 visible hake heads were identified by an observer. Assuming this figure as ground truth, the network correctly identified 175 hake heads, which represents a success rate of 87%. Concerning the false positives, two cases were detected (1%). Some examples of the output of the network are displayed in Figure 5.

Regarding the measurement performance of the system (accuracy and precision attained when estimating the fish length itself), the relationship between HL_{cm} and TL showed that the four linear models considered in the previous section (either using log-transformed values or not), had an excellent explanatory power, with r (Pearson correlation coefficient) larger than 0.9 (remark that throughout the article the terms bias and (in)accuracy and the terms variability and (im)precision are used interchangeably). However, $\log(TL)$ vs. $\log(HL_{cm})$ was finally selected because it showed the smallest deviance information criterion and normally distributed residuals. All the parameters of the model have been successfully estimated (Table 1) using the Bayesian approach described in the previous section.



Figure 5. Some segmentation results.

Table 1. Median and 95% Bayesian credibility interval of the posterior distribution for all the model parameters.

Parameters	2.5%	Median	97.5%	\hat{R}	N_{eff}
α_{body}	1.387	1.468	1.548	1.001	30 000
β_{body}	0.955	0.998	1.041	1.001	30 000
β_{head}	0.109	0.110	0.112	1.001	30 000
σ_{TL}	0.080	0.088	0.098	1.001	30 000
σ_{HLcm}	0.082	0.168	0.266	1.001	10 000
σ_{HLPix}	2.724	3.153	3.674	1.001	16 000

Values of \hat{R} close to 1 denote convergence of the MCMC chains. N_{eff} is a measure of the effective sample size on the posterior distribution. Note that σ values are at different scales and are not directly comparable.

For assessing the accuracy of the measures at the fish level (degree of closeness of estimates of a quantity to that quantity's true value), TL_{true} of 55 fish ranging from 20 to 27.5 cm (i.e. the actually measured length) was compared with TL_{est} , the value estimated from HLPix by the model. The obtained results are displayed in Figure 6. The root-mean-square deviation (RMSD)

was 1.7 cm, and the median of the unsigned deviations was 1.1 cm, suggesting that the system is accurate. However, precision (dispersion of predicted values for a given observed value) should be improved because the averaged interquartile range was 10.0 cm. Note that one random repeated measure of HLPix was used for assessing the precision and that the uncertainty at the three levels considered (posture-related error when measuring HLPix, imperfect relationship between HLPix and HLcm, and imperfect relationship between HLcm and TL) has been properly propagated. Thus, this precision estimate is the expected when a new value of HLPix will be used for estimating TL.

Finally, the system performance for estimating TL from HLPix at the box level was assessed using ten new fish boxes sampled at the auction centre (Figure 4). Two independent samples of fish from each box were used for estimating HLPix and manually measured (observer) to obtain TL_{true} . Again, the total fish length was estimated from HLPix using the model described in the previous section. The observed vs. estimated box-level mean fish lengths are shown in Figure 7. In that case, RMSD was 1.9 cm, the median of the unsigned deviations was 0.5 cm, and the maximum

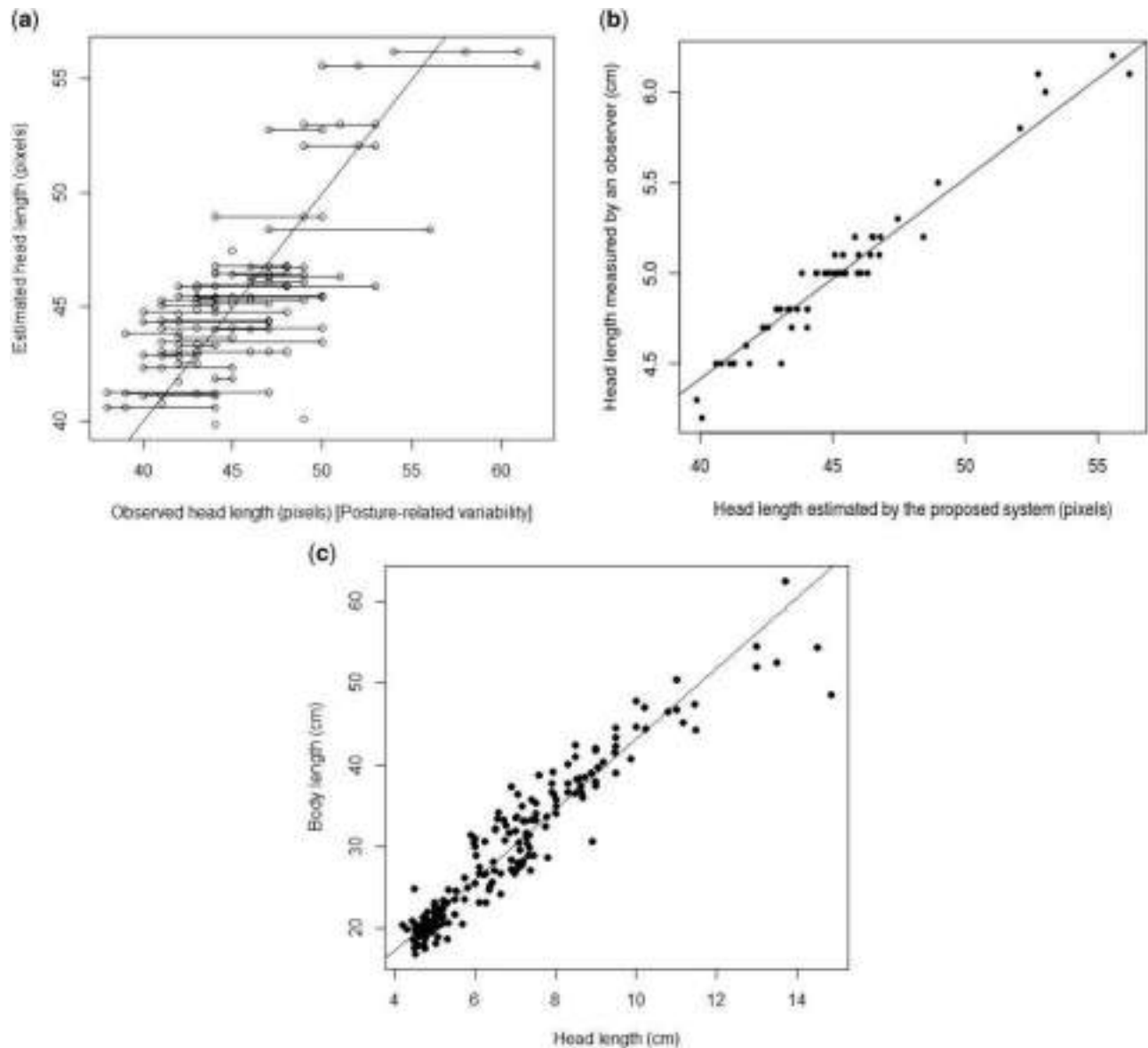


Figure 6. Variability at the three levels of uncertainty considered: (a) posture-related variability of the repeated measures of HLPix, horizontal lines connecting the repeated measures of the same fish head; (b) relationship between HLPix and HLcm; and (c) relationship between HLcm and TL.

deviation reported was 4.0 cm, suggesting that the system is accurate at the box level too.

Discussion

In line with many other successful applications of deep learning in a wide range of domains, in this study, we implemented an automatic system that uses images of captured fish at landing for identifying fish heads on those images and estimating fish length from head length.

The core component of the system is a deep neural network that permits to detect and delineate the contour of the objects of interest, or instances in the deep learning jargon. Here, instead of developing a new network from scratch, a pre-trained Mask R-CNN network was successfully implemented for identifying hake heads. This strategy implies that the network training must be fine-tuned with a relatively large data base of examples of hake

heads. In this case, the contours of 2.602 hake heads have been manually segmented from images.

The performance of the Mask R-CNN network implemented in this way for detecting hake heads is noteworthy. The majority of the heads in an image are properly detected (87%) but more interestingly for the specific case of study here, the ratio of false positives is negligible (1%).

The specificities of the case prevent whole fish contours from being efficiently detected on the images, as fish tails are systematically occluded when fishermen prepare the fish boxes. Certainly, other species are not sorted in this careful way, but even when the contour of the whole fish is visible on an image, the flexible nature of fish would complicate the performance of the detection step for the Mask R-CNN because the neural network should be taught with differently bent fish. Conversely, the rigid nature of fish heads alleviates this problem but introduces a new handicap

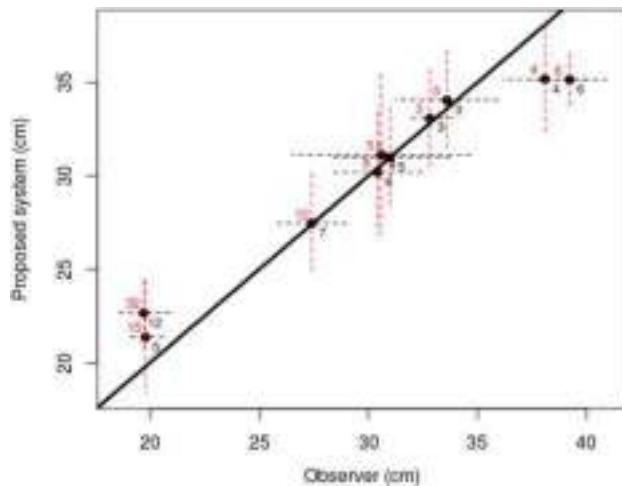


Figure 7. Observed vs. estimated box-level mean fish length. Each point represents the mean of fish length of a sample of fish in that box. The dashed lines around a point denote the between-fish standard deviation in each box. The numbers denote the sample size (number of fish) used for estimating those mean and standard deviation. Note that the fish measured are not necessarily the same fish detected by the network; thus, sample sizes may differ. The thick line denotes perfect agreement between observed and estimated fish length.

because the final objective here is not to detect fish heads but to estimate fish size. The former output of Mask R-CNN (segmentation mask of the pixels belonging to a given head) was first used to extract head length in pixels. The second step was to transform this head measure from pixels to cm and the final step was to infer fish length from head length.

Provided that each one of these steps may introduce some uncertainty, a validation protocol has been implemented for assessing the overall performance of the system in terms of accuracy and precision, as well as for providing a reliable confidence interval for fish length estimates when new measures of head length were provided by the net. A sample of fish has been measured by an observer with a ruler and these empirical measurements have been compared with the estimates provided by the system developed here. The median of the unsigned measured-estimated differences was 1.1 cm at the fish level and 1.9 cm at the box level (i.e. mean fish length), suggesting that the system should be considered accurate at least at the mid-range of the considered sizes.

However, our results show that there is room for improving the precision of the system. Individual level precision (measured as the interquartile range) for a newly measured fish length was around ± 10.0 cm for fish in the range of 20–27.5 cm. The estimates of the standard deviation related to the three variability sources considered suggest that they are similarly contributing to this suboptimal precision at the fish level. Uncertainty related to the head posture may be specially relevant. It is plausible that precise delineation of fish head contour may depend on the fish posture, thus increasing the number of examples of heads in different postures when training the network may alleviate the problem.

The morphometric relationship between head size and fish length reported in this study shows larger uncertainty than the one reported elsewhere for the same species and similar length range [Pearson product-moment correlation coefficient, $r = 0.92$

in our study and $r = 0.95$ and 0.97 in Šantić *et al.* (2011) and Philips (2014)]. Certainly, these contributions suggest some sex-related effects, which have not been accounted for in the current context.

As stated earlier, conventional, observer-based assessment of fish length is precise at fish level but, due to the inherent high costs of manual sampling, sample size will be by far smaller than the massive sample size that can be potentially processed using deep learning. Proper comparison of the effects of using observer-based data vs. deep learning data when assessing fish stock dynamics is out of the scope of this contribution but certainly deserves further attention.

The hierarchical Bayesian framework proposed here is not only appropriate for providing reliable confidence intervals at fish level. Moreover, it can be expanded for properly propagating such a fish level uncertainty to the fish box level, the boat level, the day level, or any other relevant scale that might be of interest in other case studies. Specifically, in the context of understanding fish population dynamics and taking informed management decisions on exploited stocks, a relatively low precision at the fish level may be largely compensated with a massive amount of data at upper scales.

It is in this context that the advancement in marine science is foreseen to be boosted in the next few years thanks to the capacity of generating massive amounts of data from automatic sensors coupled to high-power computation capabilities (Danovaro *et al.*, 2017; Lowerre-Barbieri *et al.*, 2019). Many techniques associated to the Artificial Intelligence are not new in marine science (e.g. simple neural networks, decision trees, and Bayesian networks). Many of these techniques are used for ecosystem modelling purposes, spatial planning, decision-making, etc. (e.g. Fernandes *et al.*, 2010). However, the field of deep learning is advancing at a greater pace, as in particular those applications related to image processing.

Until now, most applications of image classification in marine ecology were semi-supervised or supervised (e.g. Marini *et al.* 2016, 2018; Díaz-Gil *et al.*, 2017). Through deep learning, we exploit the structural characteristics of data and make use of computation capabilities (Hu *et al.*, 2014), which, in our case study, may offer a better performance than other, more conventional ways of data extraction. We clearly demonstrate that by using this method, even when only a percentage of the fish in each box can be correctly identified, opens the opportunity to massive fish length sampling of many commercially valuable species, without interfering with wharf or fishing operations and activity. Provided that an image of each fish box can be easily obtained and stored when the conveyor belt stops for bidding, the estimate number of pictures (each box) per day that currently are arriving to our system are in the order of thousands. This knowledge may enable to improve the current biological evaluation models based in size, to explore short-term effects of the environment on the species, the control of undersized individuals, or even the analysis of price dynamics within the season in relation to size. To this end, we have detected a very positive attitude from the fishery sector. Both fishermen associations and the wharf owners have facilitated and supported the initiative for extracting lengths automatically from boxes. This suggests that further development of these techniques in the near future is guaranteed. According to the above, several near-future improvements are envisaged, including (i) the detection of different species in the same image (some boxes contain a mixture of species), (ii) the automatic

calibration of cameras for the conversion from pixel unit lengths to centimeters, and (iii) an improvement of the precision of the estimation of total fish lengths from pelvic lengths.

Supplementary data

Supplementary material is available at the *ICESJMS* online version of the manuscript.

Acknowledgements

This work has been funded by the projects FOTOPEIX and FOTOPEX2 (2017/2279 and 2018/2002) from Fundació Biodiversidad, through the Pleamar Program. We specially thank OPMALLORCAMAR and Direcció General de Pesca del Govern de les Illes Balears for supporting these projects. The work of J-LL was partially supported by grants TIN2017-85572-P and DPI2017-86372-C3-3-R (MINECO/AEI/FEDERUE). This is a contribution of the Unitat Associada IMEDEA-LIMIA.

References

- Abdullah, N., Shafry, M., Rahim, M., and Amin, I. M. 2009. Measuring fish length from digital images (FileDi). *In* Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human, pp. 38–43. ACM.
- Al-Jubouri, Q., Al-Nuaimy, W., Al-Tae, M., and Young, I. 2017. An automated vision system for measurement of zebrafish length using low-cost orthogonal web cameras. *Aquacultural Engineering*, 78: 155–162.
- Arnarson, H., Bengoetxea, K., and Pau, L. 1991. Vision applications in the fishing and fish product industries. *International Journal of Pattern Recognition and Artificial Intelligence*, 2: 657–671.
- Chen, G., Sun, P., and Shang, Y. 2017. Automatic fish classification system using deep learning. *In* 2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 24–29.
- Costa, C., Loy, A., Cataudella, S., Davis, D., and Scardi, M. 2006. Extracting fish size using dual underwater cameras. *Aquacultural Engineering*, 35: 218–227.
- Dalal, N., and Triggs, B. 2005. Histograms of oriented gradients for human detection. *In* 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 1, pp. 886–893.
- Danovaro, R., Aguzzi, J., Fanelli, E., Billett, D., Gjerde, K., Jamieson, A., Ramirez-Llodra, E., et al. 2017. An ecosystem-based deep-ocean strategy. *Science*, 355: 452–454.
- d'Armengol, L., Castillo, M. P., Ruiz-Mallén, I., and Corbera, E. 2018. A systematic review of co-managed small-scale fisheries: social diversity and adaptive management improve outcomes. *Global Environmental Change*, 52: 212–225.
- Díaz-Gil, C., Sme, S. L., Cotgrove, L., Follana-Berná, G., Hinz, H., Martí-Puig, P., Grau, A., et al. 2017. Using stereoscopic video cameras to evaluate seagrass meadows nursery function in the Mediterranean. *Exported*, 164: 137. <https://app.dimensions.ai> (last accessed 3 May 2019).
- FAO. 2016. The State of Mediterranean and Black Sea Fisheries. Technical Report, FAO.
- FAO. 2018. The State of World Fisheries and Aquaculture (SOFIA). Technical Report, FAO.
- Fernandes, J. A., Irigoien, X., Goikoetxea, N., Lozano, J. A., Inza, I., Pérez, A., and Bode, A. 2010. Fish recruitment prediction, using robust supervised classification methods. *Ecological Modelling*, 221: 338–352.
- French, G., Fisher, M., Mackiewicz, M., and Needle, C. 2015. Convolutional neural networks for counting fish in fisheries surveillance video. *In* Proceedings of the Machine Vision of Animals and Their Behaviour (MVAB), pp. 7.1–7.10. BMVA Press.
- French, G., Mackiewicz, M., Fisher, M., Holah, H., Kilburn, R., Campbell, N., and Needle, C. 2019. Deep neural networks for analysis of fisheries surveillance video and automated monitoring of fish discards. *ICES Journal of Marine Science*, 77: 1340–1353.
- Hardin, R. W. 2006. Vision system monitors fish populations. *Vision Systems Design*, 11: 43–45.
- He, K., Gkioxari, G., Dollár, P., and Girshick, R. 2017. Mask R-CNN. *In* 2017 IEEE International Conference on Computer Vision (ICCV), pp. 2980–2988.
- Hilborn, R. 2007. Managing fisheries is managing people: what has been learned? *Fish and Fisheries*, 8: 285–296.
- Hu, H., Wen, Y., Chua, T., and Li, X. 2014. Toward scalable systems for big data analytics: a technology tutorial. *IEEE Access*, 2: 652–687.
- Iudicello, S., Weber, M. L., and Wieland, R. 2012. *Fish, Markets, and Fishermen: The Economics of Overfishing*. Island Press, Washington, DC.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. 2012. ImageNet classification with deep convolutional neural networks. *In* Proceedings of the 25th International Conference on Neural Information Processing Systems—Volume 1, NIPS'12, pp. 1097–1105. Curran Associates Inc., USA.
- Kruschke, J. 2010. *Doing Bayesian Data Analysis: A Tutorial Introduction with R*. Academic Press, Cambridge, MA, USA.
- Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86: 2278–2324.
- Lin, T., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. 2017. Feature pyramid networks for object detection. *In* 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 936–944.
- Lowerre-Barbieri, S. K., Catalán, I. A., Frugård Opdal, A., and Jørgensen, C. 2019. Preparing for the future: integrating spatial ecology into ecosystem-based management. *ICES Journal of Marine Science*, 76: 467–476.
- Marini, S., Azzurro, E., Coco, S., Del Rio, J., Enguádanos, S., Fanelli, E., Nogueiras, M., et al. 2016. Automatic fish counting from underwater video images: performance estimation and evaluation. *In* 7th International Workshop on Marine Technology.
- Marini, S., Fanelli, E., Sbragaglia, V., Azzurro, E., Del Rio Fernandez, J., and Aguzzi, J. 2018. Tracking fish abundance by underwater image recognition. *Scientific Reports*, 8: 13748.
- Martinez-de Dios, J. R., Serna, C., and Ollero, A. 2003. Computer vision and robotics techniques in fish farms. *Robotica*, 21: 233–243.
- Maynou, F., Morales-Nin, B., Cabanellas-Reboredo, M., Palmer, M., García, E., and Grau, A. 2013. Small-scale fishery in the Balearic Islands (W Mediterranean): a socio-economic approach. *Fisheries Research*, 139: 11–17.
- Meng, L., Hirayama, T., and Oyanagi, S. 2018. Underwater-drone with panoramic camera for automatic fish recognition based on deep learning. *IEEE Access*, 6: 17880–17886.
- Miranda, J. M., and Romero, M. 2017. A prototype to measure rainbow trout's length using image processing. *Aquacultural Engineering*, 76: 41–49.
- Monkman, G. G., Hyder, K., Kaiser, M. J., and Vidal, F. P. 2019. Using machine vision to estimate fish length from images using regional convolutional neural networks. *Methods in Ecology and Evolution*, doi: 10.1111/2041-210X.13282.
- Morales-Nin, B., Grau, A., and Palmer, M. 2010. Managing coastal zone fisheries: a mediterranean case study. *Ocean & Coastal Management*, 53: 99–106.
- Palmer, M., Quetglas, A., Guijarro, B., Moranta, J., Ordines, F., and Massutí, E. 2009. Performance of artificial neural networks and


- discriminant analysis in predicting fishing tactics from multispecific fisheries. *Canadian Journal of Fisheries and Aquatic Sciences*, 66: 224–237.
- Palmer, M., Tolosa, B., Grau, A., Mar Gil, C. d., Obregón, M., and Morales-Nin, B. 2017. Combining sale records of landings and fishers knowledge for predicting métiers in a small-scale, multi-gear, multispecies fishery. *Fisheries Research*, 195: 59–70.
- Petrell, R., Shi, X., Ward, R., Naiberg, A., and Savage, C. 1997. Determining fish size and swimming speed in cages and tanks using simple video techniques. *Aquacultural Engineering*, 16: 63–84.
- Philips, A. E. 2014. Comparison of some biological aspects between the two sexes of the European hake *Merluccius merluccius* from the Egyptian Mediterranean waters. *The Egyptian Journal of Aquatic Research*, 40: 309–315.
- Plummer, M., Best, N., Cowles, K., and Vines, K. 2006. CODA: convergence diagnosis and output analysis for MCMC. *R News*, 6: 7–11.
- Qin, H., Li, X., Liang, J., Peng, Y., and Zhang, C. Recent Developments on Deep Big Vision. 2016. Deepfish: accurate underwater live fish recognition with a deep architecture. *Neurocomputing*, 187: 49–58.
- Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. 2016. You only look once: unified, real-time object detection, *Proceedings IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788.
- Reglero, P., and Morales-Nin, B. 2008. Relationship between first sale price, body size and total catch of trammelnet target species in Majorca (NW Mediterranean). *Fisheries Research*, 92: 102–106.
- Ren, S., He, K., Girshick, R., and Sun, J. 2015. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39: 769–779.
- Šantić, M., Rađa, B., Paladin, A., and Čurić, A. 2011. Biometric properties of the European hake, *Merluccius merluccius* (Osteichthyes: Merlucciidae), from the central Adriatic Sea. *Archives of Biological Sciences*, 3: 259–267.
- Strachan, N. 1993. Length measurement of fish by computer vision. *Computers and Electronics in Agriculture*, 8: 93–104.
- Tillett, R., McFarlane, N., and Lines, J. 2000. Estimating dimensions of free-swimming fish using 3d point distribution models. *Computer Vision and Image Understanding*, 79: 123–141.
- Villon, S., Mouillot, D., Chaumont, M., Darling, E. S., Subsol, G., Claverie, T., and Villéger, S. 2018. A deep learning method for accurate and fast identification of coral reef fishes in underwater images. *Ecological Informatics*, 48: 238–244.
- Viola, P., and Jones, M. J. 2004. Robust real-time face detection. *International Journal of Computer Visio*, 57: 137–154.
- White, D., Svellingen, C., and Strachan, N. 2006. Automated measurement of species and length of fish by computer vision. *Fisheries Research*, 80: 203–210.
- Zion, B., Alchanatis, V., Ostrovsky, V., Barki, A., and Karplus, I. 2007. Real-time underwater sorting of edible fish species. *Computers and Electronics in Agriculture*, 56: 34–45.

Handling editor: Cigdem Beyan

Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Original Article

Deep neural networks for analysis of fisheries surveillance video and automated monitoring of fish discards

Geoff French ^{1*}, Michal Mackiewicz¹, Mark Fisher¹, Helen Holah², Rachel Kilburn², Neil Campbell², and Coby Needle²

¹School of Computing Sciences, University of East Anglia, Norwich Research Park, Norwich, NR4 7TJ, UK

²Marine Laboratory, 375 Victoria Road, Aberdeen, AB11 9DB, UK

*Corresponding author: tel: +44 (0)1603 592280; e-mail: gfrench@uea.ac.uk.

French, G., Mackiewicz, M., Fisher, M., Holah, H., Kilburn, R., Campbell, N., and Needle, C. Deep neural networks for analysis of fisheries surveillance video and automated monitoring of fish discards. – ICES Journal of Marine Science, 77: 1340–1353.

Received 8 May 2019; revised 28 June 2019; accepted 4 July 2019; advance access publication 1 August 2019.

We report on the development of a computer vision system that analyses video from CCTV systems installed on fishing trawlers for the purpose of monitoring and quantifying discarded fish catch. Our system is designed to operate in spite of the challenging computer vision problem posed by conditions on-board fishing trawlers. We describe the approaches developed for isolating and segmenting individual fish and for species classification. We present an analysis of the variability of manual species identification performed by expert human observers and contrast the performance of our species classifier against this benchmark. We also quantify the effect of the domain gap on the performance of modern deep neural network-based computer vision systems.

Keywords: computer vision and CCTV, deep learning

Introduction

The quantity of fish discards on-board fishing trawlers is currently estimated via measurements obtained during on-board observer sampling. The quantity of discard data is therefore limited by the availability and cost of the observers. In contrast, more precise measurements of the quantity of catch landed at port are available as it is weighed to ensure compliance with the trawlers individual quota. Quota is assigned according to the total allowable catch quota established by the Common Fisheries Policy of the European Union.

A pilot catch quota management scheme (CQMS) in the UK aimed to improve the quality of discard estimations by installing electronic monitoring systems on-board participating trawlers within the Scottish demersal fishing fleet. These systems included video surveillance cameras monitoring the conveyor belts on which fish are processed or discarded. Marine Scotland Science analysts reviewed the numbers, sizes, and species of fish caught

per vessel by sampling each vessel's video record when it returned to port (Needle *et al.*, 2014). Manually counting, measuring and identifying the species of the discarded fish has proved to be laborious and time consuming, motivating the development of a computer vision system designed to analyse the footage automatically.

The intended end result of the project is a system that supports the experts by automating as much of the tedious and expensive manual analysis as possible. We can therefore outline the main requirements of the computer vision component of the system; first to detect and count fish leaving the discard chute and second to classify and measure a subset of commercial species. Such a system must be robust to the multiple occlusions and unstructured scenes that arise in the unconstrained environment of a commercial fishing trawler; fish are randomly oriented and frequently occlude one another and the view of the working area may be occluded by fishers processing the catch (see [Figure 1](#)).

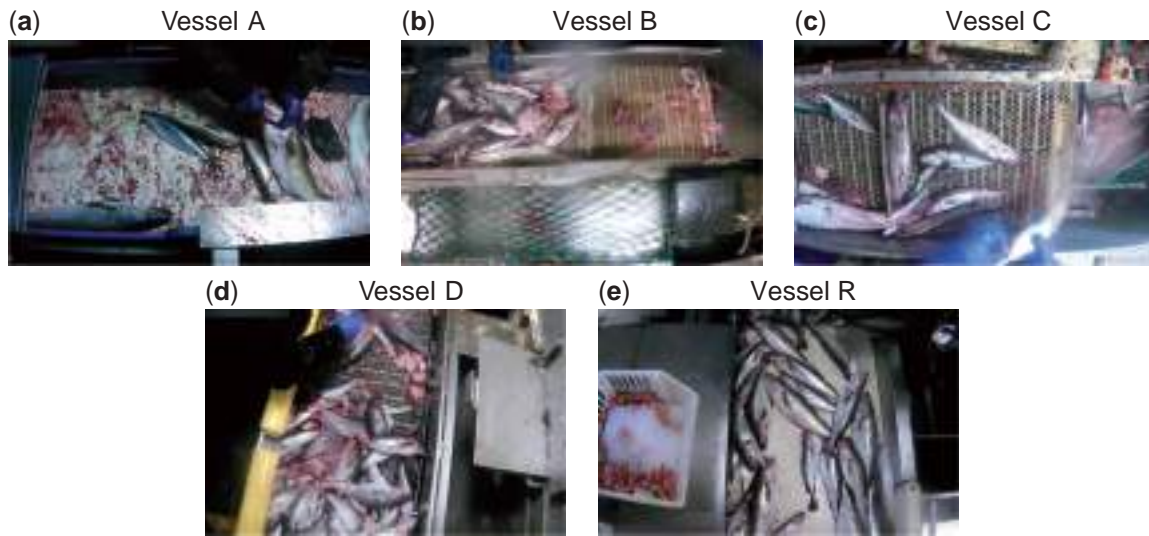


Figure 1. Images from each vessel. (a) Vessel A, (b) Vessel B, (c) Vessel C, (d) Vessel D, and (e) Vessel R.

Deep neural networks have established state-of-the-art results in computer vision problems including image classification, object detection, and image segmentation. Their impressive performance however comes at the cost of requiring large quantities of annotated training data (Lin *et al.*, 2014; Russakovsky *et al.*, 2015).

We review a body of prior work in the “Background” section. We discuss prior work in automated analysis of fishing data and work that underpins the computer vision components of our system.

The experiments that we performed required a body of training data consisting of images extracted from the video footage along with precise ground truth annotations. The dataset was developed in collaboration with observer experts at Marine Scotland, using a web-based annotation tool developed for this task. The dataset and the tool are described in the “Dataset and data acquisition tools” section.

We use instance segmentation to isolate individual fish within an image. This component is discussed in the “Instance segmentation” section. We refer our earlier work in French *et al.* (2015) that focuses on segmentation and discuss the more modern Mask R-CNN (He *et al.*, 2017) instance segmentation approach that we have adopted in its place. The fish that are detected and isolated by the segmentation system are passed to a species classifier for identification. The development of this classifier and its performance is discussed in the “Species identification” section.

To assess the performance of our classifier we conducted an experiment in which 8 expert human observers were asked to identify the species of 250 fish that were extracted from the surveillance footage. We analyse the variability of expert human observers and contrast the performance of our classifier against this benchmark in the “Inter-observer variability experiment” section.

The future directions of this work can be found in the “Conclusions and future work” section.

Background

In this section, we discuss the computer vision research that we consider relevant from the point of view of addressing our

objectives. We will specifically refer to the requirements we set out in the “Introduction” section.

Computer vision for fish classification

The first attempts to apply computer vision to the problem of fish classification were reported in the 1980s by Tayama *et al.* (1982), who used shape descriptors derived from binary silhouettes to discriminate between 9 fish species with 90% accuracy. Further work combined colour and shape descriptors (Strachan, 1993) achieving a reliability of 100% and 98% in identifying 23 species under laboratory conditions. It involved a mechanical feeding system to ensure that individual fish are correctly oriented and presented to the camera one-by-one, along with tightly controlled lighting. The author notes potential caveats due to seasonal changes in the physical condition of fish and variability in the colour of individual specimens, depending to some extent on the area in which they are caught. This issue is highly likely to affect our system too.

Further work refined approaches for fish species classification using primarily shape and colour features with fuzzy classifiers and neural networks (Hu *et al.*, 1998; Storbeck and Daan, 2001; Alsmadi *et al.*, 2009). White *et al.* (2006) describe trials of CatchMeter; a sorting machine capable of measuring and classifying fish based on colour and shape features that achieves fish length measurement accuracy of $\sigma = 1, 2$ mm and species classification accuracy of flat- and round-fish of $\sim 99\%$. Specimens must be presented individually, but can be in any orientation.

Later research investigates colour, shape, and texture features and more advanced classifiers but still requiring constrained environments avoiding occlusion. As a consequence, counting individuals is trivial or irrelevant (Hu *et al.*, 2012). However, a recent review of computer vision in aquaculture and processing of fish products identifies a wide range of applications for the technology at all stages of production (Mathiassen *et al.*, 2011; Zion, 2012), many of which present challenging problems for computer vision.

Successfully classifying images captured in real-life conditions requires the use of more sophisticated approaches such as non-rigid part models (Chuang *et al.*, 2016). Deep neural network-

based feature extractors have been successfully employed for fish species identification on the Fish4Knowledge (Boom *et al.*, 2012), using unsupervised learning to initialize the network layers (Qin *et al.*, 2016; Sun *et al.*, 2016). More recent work employs deep neural network image classifiers trained in an end-to-end fashion (Zheng *et al.*, 2018), tackling a challenging Kaggle dataset in which equipment and personnel are present in the images, in addition to the fish.

Image classification

In recent years deep neural networks have set a number of state-of-the-art image classification results. A variety of architectures have been proposed (Krizhevsky *et al.*, 2012; Simonyan and Zisserman, 2015) with residual networks (He *et al.*, 2016) combining strong performance with computational efficiency.

Practitioners frequently employ transfer learning (Donahue *et al.*, 2014; Long *et al.*, 2015) in which a pre-trained ImageNet classifier (e.g. a residual network) is adapted for a new classification task by replacing the final layer and fine tuning.

It is worth noting that deep neural networks are prone to overfitting (Krizhevsky *et al.*, 2012) and will often exhibit poor performance on data drawn from a different distribution to that on which they are trained. It is for this reason that it is important to maximize the diversity of the training set by using as wider variety of lighting and image capture conditions as possible. In situations where the annotated training images and evaluation images are drawn from different distributions or sources, the difference between them is referred to as the *domain gap*. In such situations we expect the network to perform poorly on the target/evaluation domain. The field of domain adaptation (Saenko *et al.*, 2010; French *et al.*, 2018) is aimed at finding solutions to these problems. Typical domain adaptation problems involve learning from annotated synthetic images and *unannotated* real-life images, with a view to maximizing performance on the real-life data. In surveillance situations where data are obtained from a number of cameras, a small domain gap can be said to exist between the cameras due to the different lighting conditions and perspective of each camera.

Instance segmentation

Image segmentation is the process by which an image is segmented into regions, often on a per-pixel basis. In this work, we focus on instance segmentation as our goal is to locate and isolate individual fish within an image. Instance segmentation algorithms can be divided into two classes based on how they tackle the problem.

The first approach combines semantic segmentation with contour detection. Semantic segmentation (Long *et al.*, 2015; Ronneberger *et al.*, 2015) classifies each pixel according to the type of object covering it (fish, conveyor belt, detritus, etc.). Multiple objects of the same class that touch or overlap will form a contiguous region, as occurs frequently in our CCTV footage when fish overlap. Contour detection (Xie and Tu, 2015) locates edges of objects that are used to guide the Watershed algorithm (Beucher and Meyer, 1993) to split these regions, separating individual objects. This was the approach adopted in our earlier work (French *et al.*, 2015). In practice this is often unreliable. False negatives in the contour predictions result in small gaps that prevent instances from being separated due to the flood-fill based approach of the Watershed algorithm. False-positive contour

detections can result in the complementary problem of over-segmentation. Our prior work had to train separate segmentation models for each conveyor belt (due to the aforementioned domain gap) and use carefully tuned post-processing to mitigate this problem.

The second approach to instance segmentation combines object detection and boundary localization. Object detection systems detect and locate objects within an image, typically predicting a bounding box and class category for each detected object. The instance level segmentation is generated by predicted object boundaries, often in the form of a mask that identifies the regions of the image that belong to the object in question. This is the approach adopted by Mask R-CNN (He *et al.*, 2017). They combine Faster R-CNN (Ren *et al.*, 2015) object detection algorithm—an accurate two-stage object detection algorithm—with a mask prediction module that predicts a low resolution mask (normally 28×28 pixels) that is scaled to fit the bounding box and identifies the parts of the image covered by the detected object.

Dataset and data acquisition tools

Marine Scotland provided us with the surveillance footage that was gathered during their CQMS pilot study (Needle *et al.*, 2014). In its raw form it was not suitable to be directly processed by our computer vision system. In this section, we discuss the process that we developed to extract usable image data from the CCTV video that could be annotated, allowing us to train and evaluate the machine learning components of our system.

We will discuss the source video material, the project web application, calibration, and segmentation dataset selection and preparation.

Video sources

The surveillance footage was captured in 800p HD resolution and stored in MPEG-4 format. The videos come from five sources; four commercial fishing vessels; and one research vessel operated by Marine Scotland. The footage from the commercial vessels captures the real-world working environment and presents challenging conditions, including occlusions by personnel working at the conveyor belt and the view being obscured by spatter on the dome that covers the camera. The footage from the research vessel is similar in terms of content and layout but provides the opportunity to capture tailor-made footage for the purpose of gathering training data.

The footage from the commercial vessels consists of the mix of species that was being processed on board the vessel at the time of capture. The footage from the research vessel was specifically produced by Marine Scotland staff by placing large numbers of fish of a known species on the conveyor belt and running it past the camera. Each video from the research vessel contains fish of a single species; this was done for the purpose of training the species classifier, discussed in the “Training data” section. The footage is summarized in Table 1. Example frames are shown in Figure 1.

Web application

To facilitate collaboration between Marine Scotland and University of East Anglia personnel, a web application was developed using the Django Framework (<https://djangoproject.com>). The website allows Marine Scotland staff to upload CCTV footage

and annotate images for training our computer vision systems (see “Image annotation” section). It was extended to support the inter-observer species identification variability experiment discussed in the “Performance evaluation” section.

Belt extraction and calibration

We simplified the task of processing the footage by extracting a region of interest covering the conveyor belt, thereby excluding equipment, people and the boat interior, as can be seen in Figure 1. We used a perspective transformation to extract the conveyor belt and transform it into rectilinear space (see Figure 2) with a constant uniform physical distance to image space ratio.

Lens distortion correction

The surveillance cameras on-board fishing vessels frequently use fish-eye lenses to increase field of view. This introduces a curved distortion to the image that complicates later stages of the system. The OpenCV library (Bradski, 2000) provides functionality for automatically estimating lens distortion parameters and removing it from images.

The lens distortion estimation algorithm within OpenCV requires that a printed checkerboard pattern is captured at various positions within the cameras field of view. Its corners are detected and their positions are used to estimate the lens distortion. We provided Marine Scotland staff with a checkerboard pattern and a procedure for capturing calibration footage on-board fishing vessels.

Extracting the checkerboard from all frames in which it is visible typically results in several hundred detections. The lens distortion estimation algorithm run-time scales in a super-linear fashion with respect to the number of detections used, failing to complete within a reasonable time. We opted to select a subset of the detections that significantly differ from one another.

We divide the image into 50×50 pixel cells and quantize the coordinates of the checkerboard corners, generating a map that specifies which cells are covered (histogram2D in the algorithm).

Table 1. Summary of video footage.

Vessels	Types	No of videos	Running time (HH:MM:SS)
Vessel A	Commercial	38	37:30:47
Vessel B	Commercial	23	22:45:41
Vessel C	Commercial	26	20:38:26
Vessel D	Commercial	25	24:26:56
Vessel R	Research	53	6:18:41
Total			

If $>22\%$ (determined by trial and error) of the cells covered by this checkerboard have not been covered by a previously selected checkerboard we add it to our selection. The algorithm is given below:

Algorithm 1 Lens estimation detection selection algorithm

```

covered ← BOOLEANARR2D(num_cells_y, num_cells_x)
selected_dets ← []
for each det ← detections do
  det_coverage ← HISTOGRAM2D(det, cell_size)
  if MEAN(det_coverage ∧ ¬covered) ≥ 22% then
    covered ← covered ∨ det_coverage
    selected_dets.APPEND(det)
  end if
end for

```

Belt warping

The checkerboard used for estimating lens distortion parameters was printed on A3 paper, giving it known physical dimensions. The checkerboard was placed on the conveyor belt and captured as part of the calibration process. The checkerboard localization algorithm within OpenCV is used to find the checkerboard, after which a perspective transformation is estimated to transform the checkerboard into a fixed rectangular size. Applying this transformation to the image of the belt removes the perspective distortion and scales the image of the belt to a known physical distance to image space ratio. A tool was developed within Jupyter Notebook (Kluyver *et al.*, 2016) that allows the user to correct for any misalignment and crop the region corresponding to the belt.

Complete belt extraction process

We use the estimated lens parameters to compute a mapping. For each pixel in the straightened image the mapping provides its coordinates in the distorted image. The perspective transformation used for belt extraction can also be expressed as a mapping. We therefore compute a composite mapping that combines both the distortion removal and perspective transformation in a single step. The composite mapping is generated once and used for each image or frame that must be processed.

The mapping can be applied to an image using GPU accelerated texture map lookups and typically takes <2 ms on a desktop machine.



Figure 2. The belt extraction and calibration process: (a) checkerboard on belt, (b) with lens distortion removed, and (c) with perspective warp used to transform belt into rectilinear space and exterior cropped out.

Segmentation and species ID training set

A segmentation dataset consisting of still frames extracted from the video footage was required to train and evaluate the segmentation system. The conveyor belt moves in irregular and unpredictable short bursts and is controlled by on-board personnel. We wished to extract frames such that the belt moves by at least half the length of the visible region of the belt to ensure that the content changes sufficiently between frames extracted for the training set. This required a robust estimate of the belt motion. We should note that there is overlap between successive frames, so some individual fish are visible in more than one training set frame.

Belt motion estimation

Extracting the belt from the image and transforming it into rectilinear space simplifies the task of estimating belt motion between frames as its motion is constrained to horizontal translation. A natural choice for this would be enhanced correlation coefficient-based image alignment (Evangelidis and Psarakis, 2008), an implementation of which is provided by OpenCV. Unfortunately this algorithm is often confused by the repeating texture present on the conveyor belts in our footage. We developed a more robust solution based on correlation of neural network features.

While inter-frame correlation between RGB or greyscale pixel data was sufficient to detect motion, it did not accurately quantify it. To precisely quantify the motion we computed the correlation between features extracted using the convolution layers of a pre-trained VGG-16 (Simonyan and Zisserman, 2015) network instead of RGB pixel values. We found that later layers of the network would yield more accurate motion estimates, but at reduced resolution.

Once correlation using RGB pixel values indicated motion, features were extracted from the pool4, pool3, and pool2 layers of VGG-16. The pool4 feature correlations provided an accurate estimate of motion, but at 1/16 resolution. Correlation between pool3 at 1/8 resolution was computed and their output constrained so as to refine the motion estimate from pool4. Further refinements were obtained using features from pool2, after which final refinements were calculated using RGB pixel value correlation.

Our implementation uses the pre-trained VGG-16 network provided by the torchvision library that is part of PyTorch (Paszke et al., 2017).

Image annotation

The images selected for segmentation were uploaded to the web application after which they were manually annotated by Marine Scotland staff. Within this application the labelling tool allows the user to draw polygonal annotations and classify them (<http://bitbucket.org/ueacomputervision/image-labelling-tool>). The user can select from 15 species of fish and several non-fish classes such as person, belt structure, or guts. There are also classes used to indicate unidentifiable fish or material. The labelling tool can be seen in Figure 3.

Manually annotating fish by drawing polygonal labels is a labour intensive task. We were able to considerably reduce the labelling effort required by partially automating this process. Once between 100 and 200 images had been manually annotated for each belt, we found that a segmentation model trained using these annotations was able to automatically annotate the majority of fish to a satisfactory standard. We generated automatic



Figure 3. Web-based segmentation annotation tool.

Table 2. Segmentation training set.

Vessels	No of annotated images	No of annotated fish
Vessel A	204	1 459
Vessel B	263	1 254
Vessel C	145	1 588
Vessel D	153	4 809
Vessel R	137	1 498
Total	902	10 608

annotations for as-of-yet unannotated images and placed them on the website to serve as a starting point for the annotators. This saved considerable effort as the annotators only needed to annotate the few fish that had been missed or fix mistakes. The improved annotations could then be added to the training set that was used to train a new and more accurate segmentation model, resulting in a cyclic process.

Data

The training data for the segmentation system consists of 902 annotated frames drawn from videos from the five vessels and is summarized in Table 2. While many more frames were extracted, this is the subset that has been annotated so far.

Instance segmentation

An effective instance segmentation algorithm is a pre-requisite to the successful operation of the complete system as later stages rely on accurate detection and segmentation to reliably classify the species of individual fish and estimate length and mass.

During the course of the project we experimented with a variety of approaches to solving this problem. Our first attempt used semantic segmentation to identify regions of the image containing fish and subsequently split them into individuals using contour detection. By using a separate segmentation model for each conveyor belt and finely tuned post-processing we were able to achieve some success using VGA resolution footage (French et al., 2015). This process proved unreliable when applied to higher resolution HD footage as false negatives from the contour detector would prevent separation of individual fish from one another.

Mask R-CNN (He et al., 2017) proved to be an effective and efficient instance segmentation algorithm, hence we adopted it for use in our system [We use the COCO (Lin et al., 2014) pre-trained implementation of Mask R-CNN provided by the torchvision library that is developed by the PyTorch (Paszke et al., 2017) team. It produces good results and trains quickly.]. As stated in



Figure 4. Instance segmentation applied to a frame from footage from Vessel R (research vessel); the outlined shapes are generated automatically.

the “Background” section, it combines object detection with mask prediction and is therefore much more robust than our previous approach. It generates high quality labels as shown in Figure 4. Furthermore our segmentation model is trained on images from all vessels simultaneously.

As stated in the “Image annotation” section, the segmentation system was used to automatically annotate images on the labelling tool section of the project web application, after which mistakes in the annotations could be fixed manually. We maximized the quality of the automatically generated annotations using test-time augmentation (He *et al.*, 2017); each image was segmented eight times, with random augmentation consisting of horizontal and vertical flips, lightening and darkening, scaling and rotation. The resulting predictions were averaged, increasing their accuracy. Doing so comes at significant computational cost, so this is only feasible for offline use when accuracy outweighs run-time performance.

Separate species identification

The object detection network that forms the basis of Mask R-CNN (He *et al.*, 2017) incorporates a classifier that identifies detected objects and a multi-class mask head that learns class-specific shapes for segmentation. In principle this could be used to perform fish detection, segmentation, and species identification in a single pass. In spite of this we opt to use separate networks, using a single class Mask R-CNN network for only fish detection and segmentation. We do this for several reasons that we will now explain.

Identifying the species of fish in our surveillance footage requires annotators with the relevant training and experience. In contrast outlining individual fish for segmentation can be performed by a wide variety of individuals. To support this we allow annotators to outline fish in an image without specifying their species. As a consequence many images in our dataset have fish outlined for segmentation but with some individuals having no assigned species. Training a multi-class Mask R-CNN model requires per-object class labels to select the class-specific bounding box regressor and mask head to optimize for each object. As a consequence images with partial species annotation would not be usable for training a multi-class Mask R-CNN model.

Furthermore, as stated in the “Classifier” section we were able to improve the performance of our classifier by rotating the images of segmented fish so that they lie horizontally, as doing so eliminates a source of irrelevant variation. Mask R-CNN does not provide a mechanism for altering the orientation of objects prior to classification.

For these reasons we train our Mask R-CNN model to detect and segment objects of a single fish class and identify species in a subsequent step.

Training procedure

Data augmentation artificially expands the training set by modifying the existing image samples to increase variability and is frequently used to improve performance (Krizhevsky *et al.*, 2012; He *et al.*, 2016). While training our segmentation network we augment the images using random horizontal and vertical flips, random rotations between -45° and 45° , applying a random uniform scale factor in the range of 0.8–1.25 and randomly modifying the brightness and contrast by multiplying the RGB values by a value drawn from $e^{\mathcal{N}(0, \ln(0.1))}$ and adding a value drawn from $\mathcal{N}(0, 0.1)$.

We split our dataset into 90% for training and 10% for validation. We train for 350 epochs with one epoch consisting of the iterations necessary to train using all training images. We report the mean average precision (mAP; Lin *et al.*, 2014) score for the validation samples in our logs. We use the validation score for early stopping; we save the network state for use after the epoch at which it achieved the highest validation mAP score. We use a learning rate of 10^{-4} for the new randomly initialized later layers and 10^{-5} for the pre-trained layers that come from the *torchvision* (Paszke *et al.*, 2017) Mask R-CNN implementation. We randomly crop 512×512 pixel regions from our rectilinear belt images and build mini-batches of crops from four randomly chosen images during training. We train our models on a single nVidia GeForce 1080-Ti GPU.

In addition to the bounding box non-maximal suppression used in Mask R-CNN (He *et al.*, 2017) we apply NMS to the masks predicted during inference. If $>10\%$ of the pixels predicted as belonging to object are already occupied by other objects with a higher predicted confidence, the lower scoring object is ignored.

Species identification

In this section we describe our species classifier, the development of the dataset required for training and our evaluation of the performance of our classifier.

Classifier

Our species classifier is a 50-layer residual network (He *et al.*, 2016) adapted and fine tuned using transfer learning. It operates on images of individual fish that are identified by the instance segmentation system (see “Instance segmentation” section).

We found careful pre-processing of images of individual fish to be essential for good classification performance. While the fish in our surveillance footage are arbitrarily oriented, we found that rotating images of individual fish so that they lie horizontally eliminated a source of irrelevant variation, improving accuracy. We used the *regionprops* function from the Scikit-Image (van der Walt *et al.*, 2014) library to estimate the orientation from the shape/mask predicted for each fish and rotate it so that the longest axis lies horizontally. This ensures that most fish lie horizontally,

although they vary in horizontal and vertical direction (left-to-right or right-to-left, upside-down). Given that the masks predicted by the segmentation system are often imperfect we found that expanding the mask in all directions by seven pixels (using binary dilation) improved performance. Each image was scaled to a constant size of 192×192 pixels and centred within a 256×256 image. Pixels outside of the masked to 0, removing any distracting cues from parts of the image outside the bounds of the fish.

Training data

Our species identification training data is drawn from footage from the commercial vessels and from the research vessel.

A summary of the species identification training data broken down by vessel and species is given in Tables 3 and 4.

The commercial training samples were drawn from commercial footage and their species was determined manually. This is a time consuming and laborious process, hence the limited amount of commercial samples, as shown in Table 3. With a view to addressing this, Marine Scotland staff prepared placed large quantities of fish of known species on the research vessel conveyor belt and ran it past the camera. Applying the segmentation system allowed us to extract large numbers of training images of a known species class, resulting in the research training samples summarized in Table 4. This further illustrates the advantage of separating segmentation and species classification into separate steps, as mentioned in the “Separate species identification” section.

The commercial training samples were extracted using manually prepared polygonal segmentation as the annotators used the labelling tool to provide both polygonal segmentation and species identification annotations for commercial images at the same time. In contrast, the majority of the research samples was extracted using boundaries generated by the segmentation system, with test-time augmentation in use. We should note that a system deployed in the field would not use test-time augmentation as segmenting each image multiple times under differing augmentation parameters incurs significant computational load. While as a consequence, a real-life species classifier would receive slightly lower quality segmentation labels than those used here, we believe that with the

Table 3. Summary of species identification dataset from commercial vessels.

Vessel	Cod	Haddock	Whiting	Saithe	Hake	Monk
Vessel A	47	109	12	370	116	1
Vessel B	21	89	25	9	9	7
Vessel C	19	229	23	70	31	2
Vessel D	12	258	42	21	16	
Total	99	685	110	470	172	10

Table 4. Summary of species identification dataset from the research vessel.

	Cod	Haddock	Whiting	Saithe	Hake	Monk	Mackerel
No of fish	1 451	12 482	14 068	861	304		1 837
No of videos	3	18	13	2	2		1
	Horse mackerel	Norway pout	Plaice	Long rough dab	Common dab	Grey gurnard	Red gurnard
No of fish	496	5 574	2 402	1 495	1 601	1 599	65
No of videos	1	2	3	1	2	3	1

increased size of the training set that we are continually growing, this should not be a significant problem in the final application.

It should be noted that the complex and unstructured scenes in our CCTV footage frequently feature fish that are oriented such that useful discriminative features or parts are hidden from view or fish that are only partially visible due to being occluded by overlapping fish or personnel working at the belt. Operating in these challenging conditions is one of the challenges posed by this project. Selected examples from each species are shown in Figure 5.

Performance evaluation

To understand the performance of our classifier we evaluate it in four scenarios. In our first scenario we train and test the classifier on research samples. Given the large number of available training samples, uniform lighting and appearance and the fact that there are typically less occlusions that in the commercial footage we expect this to provide an upper bound for the performance of our classifier. In our second scenario we train and test using commercial samples. There are considerably less training samples available and the conditions are more challenging so we expect our classifier to overfit the training data to a greater extent and exhibit worse performance. We also add the research samples to the training set to assess their effect. In our third scenario we use leave-one-belt-out cross validation to test on samples from one commercial belt and train on samples from the other commercial belts and the research samples. This scenario is more representative of a system deployed in the field that must operate on samples from a belt that was not in the training set. In our final scenario we train on research samples and test on commercial samples. This is by far the most challenging scenario for the classifier due to the domain gap between the research and commercial belts. It is also the ideal scenario from the perspective of preparing training data due to the reduced annotation effort.

In scenarios in which samples from one or more belts are used for both training and testing we split the samples between train and test using fourfold cross validation. As stated in the “Segmentation and species ID training set” section individual fish may be seen in multiple successive frames extracted from video footage. We split samples using the video from which they were drawn (all the samples from a video are placed into either train or test), ensuring that a sample cannot appear in both the training and the test set.

We present the performance of our classifier using a confusion matrix. Each row of the matrix shows the distribution of how samples of that class were predicted and mis-predicted by the classifier. The values along the diagonal give the class accuracies; the proportion of samples belonging to a class that are correctly identified by the classifier. Other entries in the same row show

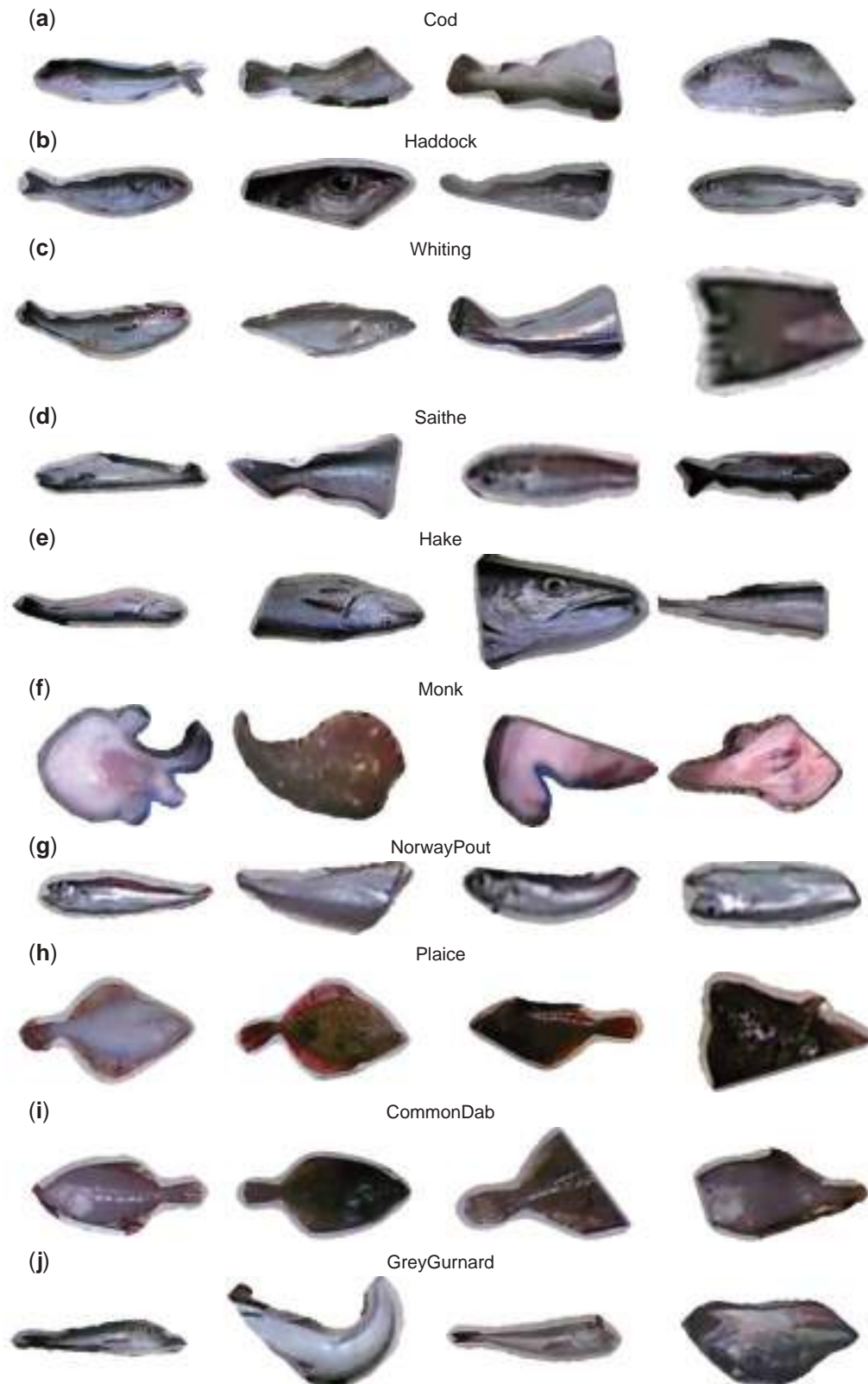


Figure 5. Examples from the species identification dataset. All fish were from the single species research vessel footage, apart from monk which were taken from commercial footage. Samples were chosen to illustrate that the classifier often receives only a partial fish or one whose orientation hides useful details.

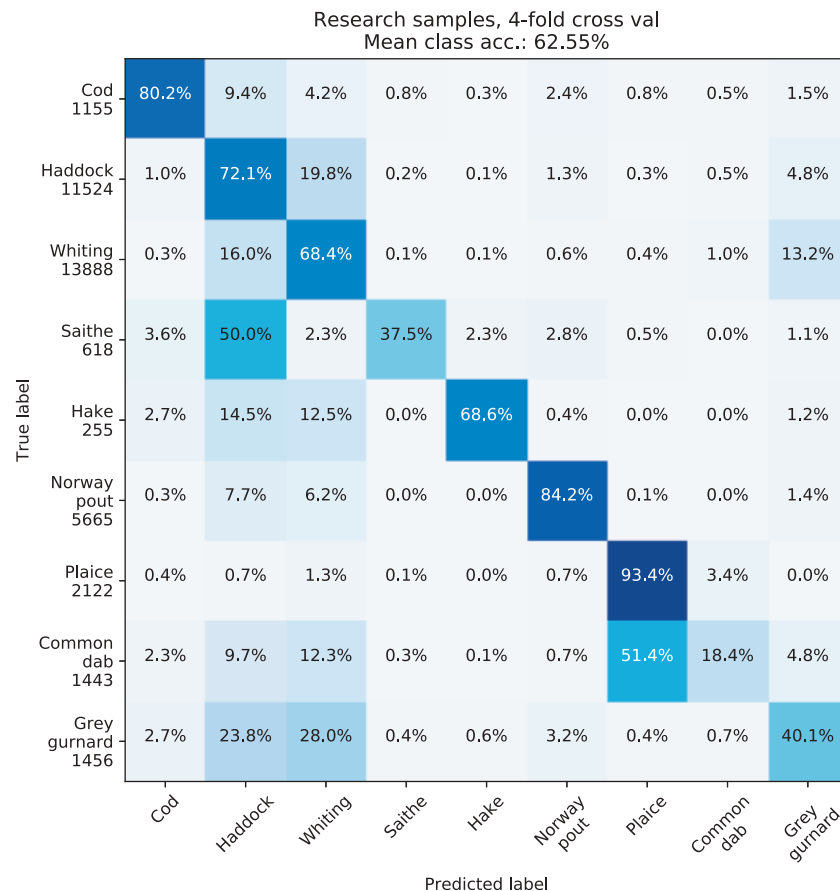


Figure 6. Confusion matrix for research samples, fourfold cross validation.

the proportion of samples mis-predicted as belonging to other classes. Perfect performance is indicated by 100% along the diagonal and 0% everywhere else.

Train and test on research samples

The research footage covers 13 species out of the 14 considered in this project. We do not consider monk as there are no examples in the research footage. We also skip mackerel, horse mackerel, long rough dab, and red gurnards as these species are only featured in one video each, preventing us from splitting the videos between train and test. When training and testing on research samples we obtain the performance shown in Figure 6. While deep neural network classifiers are effective, problems can arise when attempting to distinguish classes that are broadly visually similar, hence saithe being mis-predicted as haddock and common dab mistaken for plaice. The distribution of the confidence predicted by the classifier does not sufficiently differ between correctly and incorrectly predicted samples to allow one to reliably estimate the correctness of a specific prediction, however the difference would suggest that confidence could be used as a signal to prioritize difficult unannotated samples for manual annotation (Wang and Shang, 2014).

Train and test on commercial samples

Figure 7a and b shows the performance obtained on commercial samples when training using (a) commercial samples and (b) both commercial and research samples. Adding the research

samples—of which there are ~20 times as many as there are commercial—incurs the risk of the classifier being dominated by the research samples. Combining these datasets initially appears to degrade performance as the mean class accuracy drops from 59.16 to 56.71%. If we ignore the monk class due to lack of representation in the research samples the mean class accuracy increases from 59 to 62.05%. Adding the research samples with its large number of examples of whiting increases class accuracy, partially compensating for the poor whiting class accuracy in (b) due to the scarcity of whiting in the commercial samples.

Leave-one-belt-out cross validation

In practice a system such as the one discussed here would need to be deployed for usage on vessels for which there is no annotated training data. To assess the potential impact on performance in practical scenarios we trained five classifiers, each one on samples from four out of five vessels, with samples from the remaining vessel held out for testing. The results are presented in Figure 8. The large variation in performance evident in (b) and (d) when evaluating on samples from Vessel B and Vessel D indicates per-belt bias in the training samples that needs to be explored further. The reduction in accuracy in comparison to that in Figure 7 illustrates the effect of the domain gap.

Train on research and test on commercial samples

The performance obtained from training with samples from research footage that contains only cod, haddock, whiting, saithe,

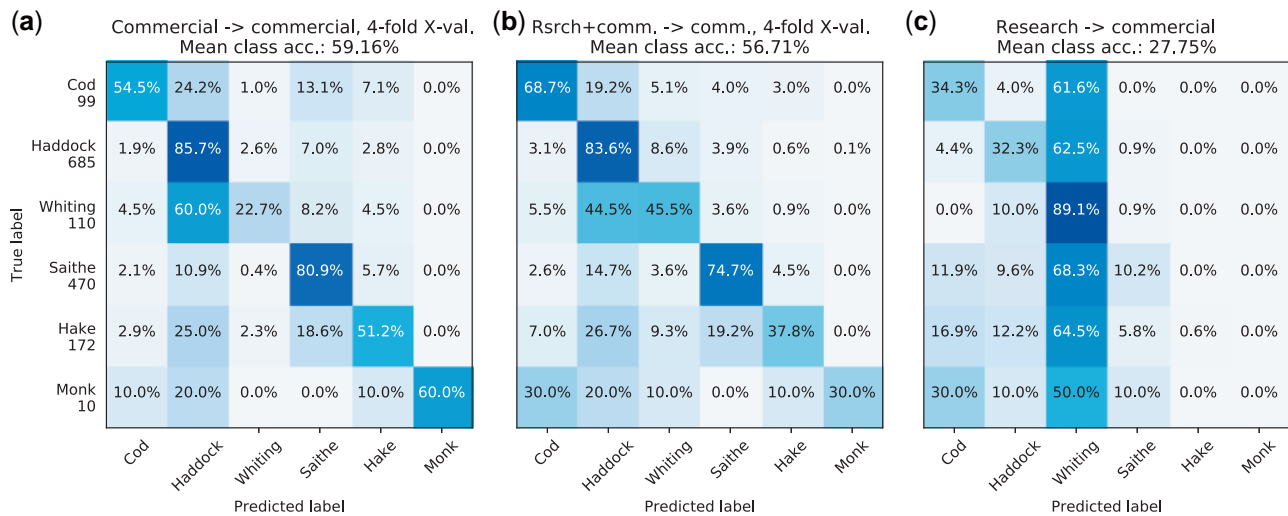


Figure 7. Confusion matrices for (a) train and test on commercial (fourfold cross validation), (b) train on research and commercial, test on commercial (fourfold cross validation), and (c) train on research, test on commercial. Without the monk class the mean class accuracies are (a) 59%, (b) 62.05%, and (c) 33.3%.

and hake and testing on the commercial samples is shown in Figure 7c. Comparing the performance between (a) and (c) illustrates the effect of the domain gap; in spite of the fact that there are ~20 times as many research samples as commercial, training using only research samples results in considerably worse accuracy, with significant numbers of samples from all classes being mis-predicted as whiting.

Inter-observer variability experiment

In this section, we describe the species identification inter-observer variability experiment that was designed to measure the accuracy of expert human observers, against which we compare the accuracy of our classifier.

Two hundred and fifty images of fish were extracted from the mixed species footage. Their background was darkened and blurred to suppress irrelevant cues and they were oriented horizontally. These images were presented to expert observers in a web-based tool—see Figure 9—that asked them to assign a species and difficulty rating to each image. The species identification tool was integrated into the project web application. It allows users to pan and zoom to focus on fine details. The user may choose a more comfortable orientation using the controls along the top to flip the image or rotate it by 180°.

We selected fish from the mixed species data as these are representative of real-world conditions. We decided that we needed at least 50 instances of each species used in the experiment to ensure sufficient representation for the purpose of meaningful analysis. Given the class imbalance present in our data (see Table 3) we used the existing species annotations to select samples for the dataset. While these individual fish had been previously annotated by Marine Scotland staff who later participated in this experiment, the samples were originally annotated in the context of a complete image including other fish, the conveyor belt and surroundings, whereas in this experiment the fish were extracted from their surroundings. The requirement of 50 samples per class prevented us from using monk in our assessment due to insufficient availability of samples. Fifty samples were selected from the

remaining five classes (cod, haddock, whiting, saithe, and hake), hence the dataset containing 250 samples.

We should note that observers from Marine Scotland reported that several samples belonged to species that could not be chosen from the five species available. Due to the fact that we did not anticipate this situation, no option indicating a different species was available, so the observers chose a combination of unidentifiable species with very easy difficulty. This issue persists in our data and would need to be corrected in future experiments.

Expert observer agreement

We present our results in the confusion matrices shown in Figure 10. Each confusion matrix compares the species choices of one observer with the majority choice of the other seven.

The expert observers are largely in agreement with one another with mean class accuracy scores ranging from 74.4 to 86%, with the exception of observer 6 with a score of 51.4% due to low scores on whiting and hake.

Comparing the classifier with expert observers

We use the majority species choice for each sample in the inter-observer variability dataset as the ground truth for evaluating three classifiers: one trained on single species samples from the research vessel, one trained on the mixed species samples from the commercial vessels and one trained on a combination of both. In each case the samples in the inter-observer variability dataset are held out as test data with other samples used for training. The results are presented in Figure 11. Following the *leave one belt out strategy* discussed in the “*Leave-one-belt-out cross validation*” section, we obtain the results in Figure 12.

The comparison between the agreement between human observers shown in Figure 10 and the performance of the classifier shown in Figure 11 show that there is a significant gap that must be crossed before human accuracy is reached, especially when crossing the domain gap as in Figure 12. Expert human observers typically score a mean class accuracy of between 74 and 86%, whereas the classifier reaches around 58%, slightly outperforming observer 6, the lowest scoring human observer.

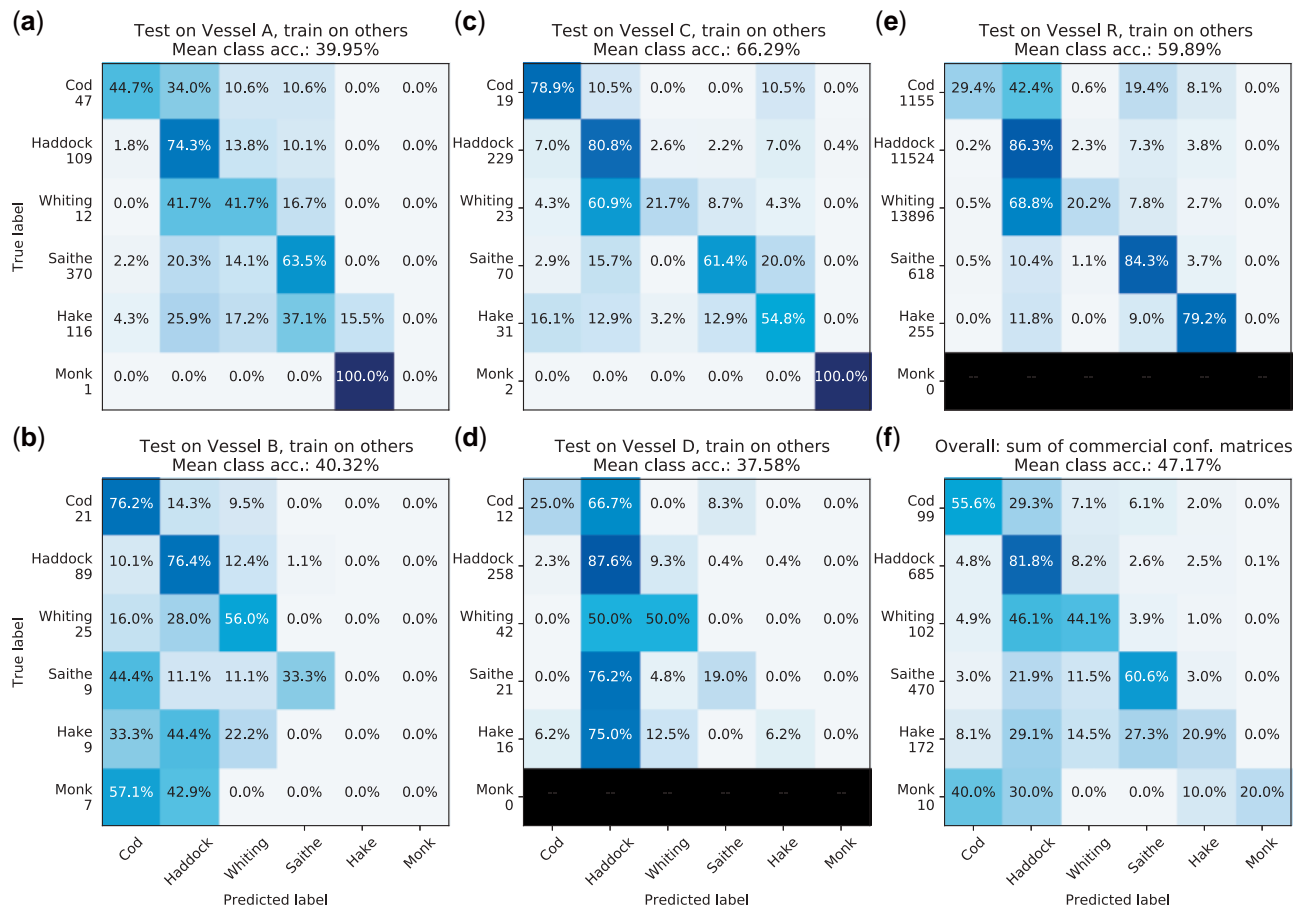


Figure 8. Performance when evaluating on samples from one vessel while training on others. Overall performance the result of computing the sum of the other confusion matrices. Overall mean class accuracy without under-represented monk class is 52.6%.



Figure 9. Inter-observer variability species identification tool as seen by the participants.

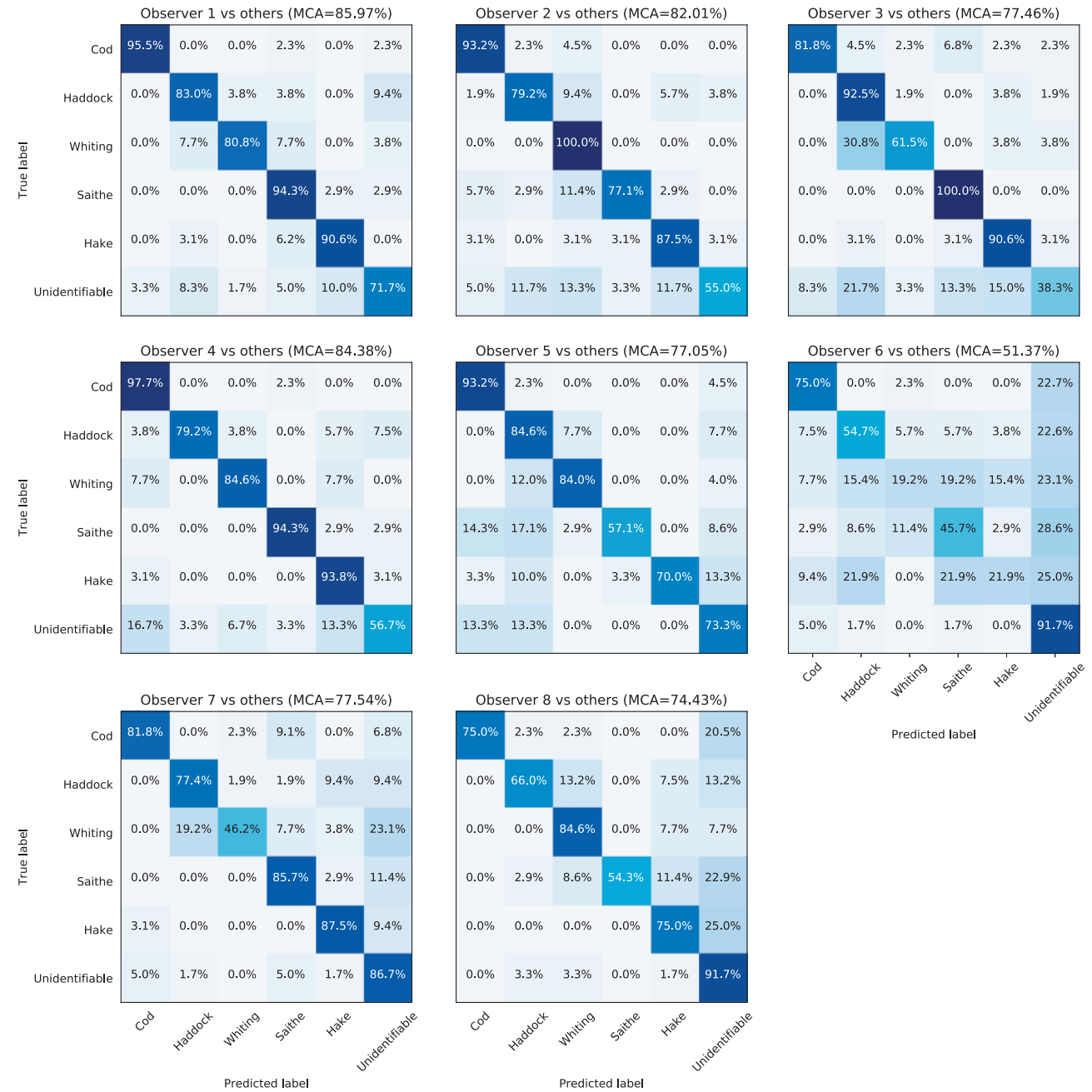


Figure 10. Inter-observer agreement confusion matrices. Each confusion matrix compares the species choice of an observer with the majority vote of the other seven observers.

Conclusions and future work

We have discussed the development of a system for analysing and quantifying fish discards from CCTV footage captured on fishing trawlers. Is designed to operate in the challenging real-world conditions present in these environments. The major components of the system are in place. The remaining challenges include length estimation, tracking fish between frames and reidentification to handle situations where fish go out of view temporarily due to occlusion. There is a significant body of work on the topic of person re-identification (Li *et al.*, 2018), some of which could be adapted to this problem.

The segmentation system is performing adequately and we believe that its performance will continue to improve as more training data is gathered.

The main outstanding challenge is improving the performance of the species classifier. The performance obtained using footage from the research vessel (shown in the “Train and test on research samples” section and Figure 6) demonstrates that effective species classification is possible given sufficient training data. Good performance on commercial samples was achieved for some species provided that training data from all belts was used (see Figure 7a and b). We believe that growing the number of annotated

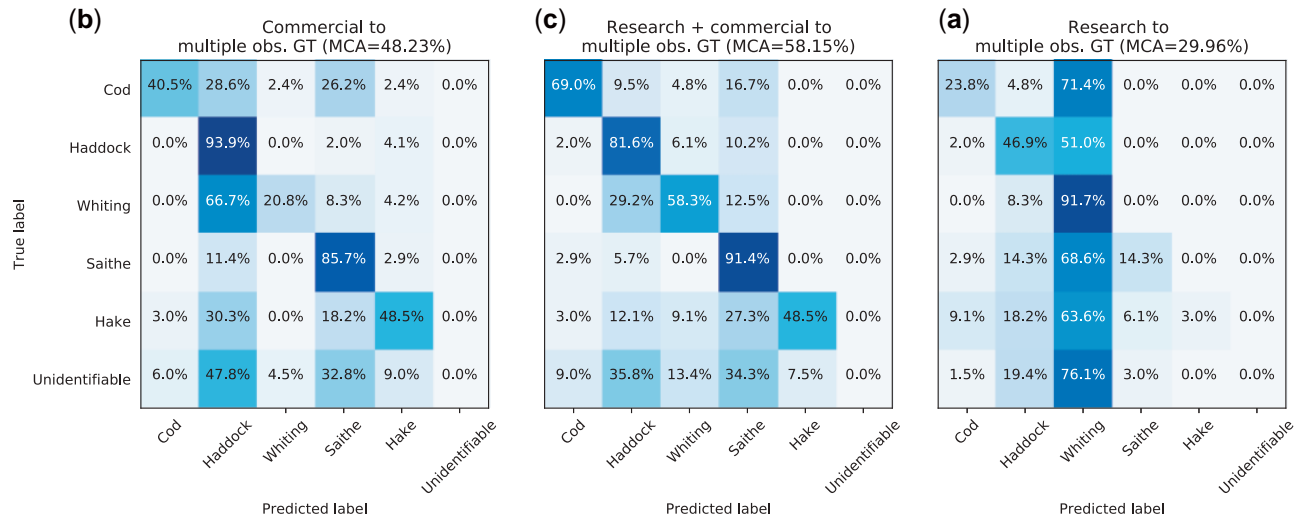


Figure 11. Classifier predictions in comparison to those of the expert observers.

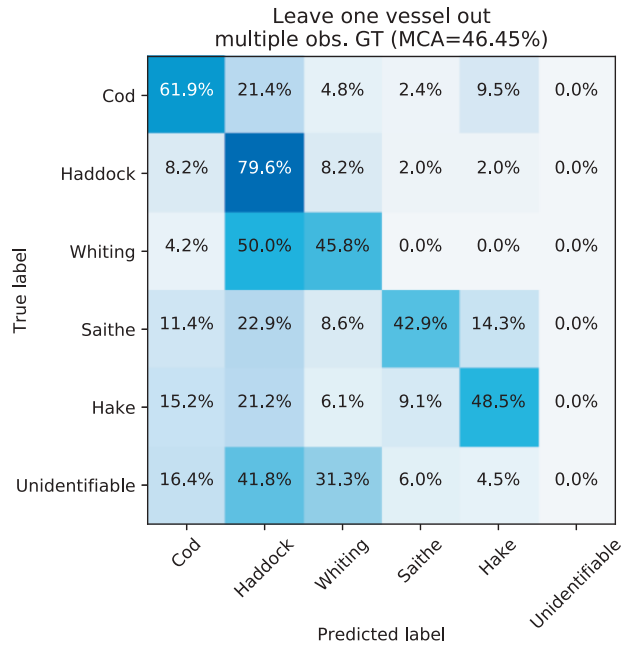


Figure 12. Classifier predictions in comparison to those of the expert observers; evaluate on samples from one vessel while training on samples from others.

commercial samples will further improve performance, reaching that of the research footage. This would however involve considerable manual effort. This effort could be supported by improving the user interface of the annotation tools. We also note that active learning offers the possibility of estimating the difficulty of unannotated samples and using it to prioritize them for manual annotation, optimizing the use of the annotators' time.

The single species research footage proved to be a highly effective approach for gathering a large number of labelled training samples in an efficient manner, although it had the disadvantage of having relatively uniform lighting and visual characteristics. The effect of the domain gap can be seen by comparing the results presented in Figure 7a and c. An avenue we intend to explore

with Marine Scotland staff involves the use of an on-shore conveyor belt that affords us the opportunity to change the belt material and appearance and modify the lighting to increase the diversity of visual characteristics expressed by the dataset. If this results in sufficient accuracy, this would support the efficient production of large quantities of annotated training samples.

Active learning offers the possibility of estimating the difficulty of unannotated samples and using it to prioritize them for manual annotation, optimizing the use of the annotators' time.

Fine-grained classification is a field of on-going research aimed at developing classifiers that can distinguish between classes of objects whose overall appearance is very similar with only subtle or small differences differentiating them. Effective fine-grained classifiers locate regions of an image—often bounding boxes—that are likely to be discriminative (Yang *et al.*, 2018; Guo and Farrell, 2019). Such classifiers could be well suited to the problem of fish species identification.

We can conclude that the use of computer vision to quantify fish discards from surveillance footage is feasible with current state-of-the-art algorithms.

Acknowledgements

We would like to thank James Dooley, Charlotte Altass, Luisa Barros, Lauren Clayton, and Anastasia Moutaftsi from Marine Scotland and Rebecca Skirrow from CEFAS for participating in our species identification inter-observer variability experiment. We would like thank nVidia corporation for their generous donation of a Titan X GPU.

Funding

This work was funded under the European Union Horizon 2020 SMARTFISH project, Grant Agreement No. 773521.

References


Alsmadi, M. K. S., Omar, K. B., Noah, S. A., and Almarashdah, I. 2009. Fish recognition based on the combination between robust feature selection, image segmentation and geometrical parameter techniques using artificial neural network and decision tree. *International Journal of Computer Science and Information Security*, 2: 215–221.

- Beucher, S., and Meyer, F. 1993. The morphological approach to segmentation: the watershed transformation. *Mathematical morphology in image processing*. Optical Engineering, 34: 433–481.
- Boom, B. J., Huang, P. X., He, J., and Fisher, R. B. 2012. Supporting ground-truth annotation of image datasets using clustering. *In Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012)*, pp. 1542–1545. IEEE.
- Bradski, G. 2000. OpenCV. Dr. Dobb's Journal of Software Tools. <https://opencv.org/>.
- Chuang, M.-C., Hwang, J.-N., and Williams, K. 2016. A feature learning and object recognition framework for underwater fish images. *IEEE Transactions on Image Processing*, 25: 1862–1872.
- Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., and Darrell T. 2014. Decaf: a deep convolutional activation feature for generic visual recognition. *In International Conference on Machine Learning*, pp. 647–655.
- Evangelidis, G. D., and Psarakis, E. Z. 2008. Parametric image alignment using enhanced correlation coefficient maximization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30: 1858–1865.
- French, G., Fisher, M., Mackiewicz, M., and Needle, C. 2015. Convolutional neural networks for counting fish in fisheries surveillance video. *In Proceedings of Machine Vision of Animals and Their Behaviour Workshop at the 26th British Machine Vision Conference*.
- French, G., Mackiewicz, M., and Fisher, M. 2018. Self-ensembling for visual domain adaptation. *In International Conference on Learning Representations*. <https://openreview.net/forum?id=rkpoTaxA->
- Guo, P., and Farrell, R. 2019. Aligned to the object, not to the image: a unified pose-aligned representation for fine-grained recognition. *In 2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pp. 1876–1885. IEEE.
- He, K., Gkioxari, G., Dollár, P., and Girshick, R. 2017. Mask R-CNN. *In IEEE International Conference on Computer Vision (ICCV)*, pp. 2980–2988. IEEE.
- He, K., Zhang, X., Ren, S., and Sun, J. 2016. Deep residual learning for image recognition. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778.
- Hu, B. G., Gosine, R., Cao, L. X., and de Silva, C. 1998. Application of a fuzzy classification technique in computer grading of fish products. *IEEE Transactions on Fuzzy Systems*, 6: 144–152.
- Hu, J., Li, D., Duan, Q., Han, Y., Chen, G., and Si, X. 2012. Fish species classification by color, texture and multi-class support vector machine using computer vision. *Computers and Electronics in Agriculture*, 88: 133–140.
- Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B. E., Bussonnier, M., Frederic, J., Kelley, K. *et al.* 2016. Jupyter Notebooks—a publishing format for reproducible computational workflows. *In 20th International Conference on Electronic Publishing*, pp. 87–90.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. 2012. ImageNet classification with deep convolutional neural networks. *In Advances in Neural Information Processing Systems 25*, pp. 1097–1105.
- Li, M., Zhu, X., and Gong, S. 2018. Unsupervised person re-identification by deep learning tracklet association. *In Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 737–753.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. 2014. Microsoft, coco: common objects in context. *In European Conference on Computer Vision*, pp. 740–755. Springer.
- Long, J., Shelhamer, E., and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440.
- Mathiassen, J. R., Misimi, E., Bondø, M., Veliyulin, E., and Østvik, S. O. 2011. Trends in application of imaging technologies to inspection of fish and fish products. *Trends in Food Science & Technology*, 22: 257–275.
- Needle, C. L., Dinsdale, R., Buch, T. B., Catarino, R. M. D., Drewery, J., and Butler, N. 2014. Scottish science applications of remote electronic monitoring. *ICES Journal of Marine Science*, 72: 1214–1229.
- Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z. *et al.* 2017. Automatic Differentiation in PyTorch. *Neural Information Processing Systems Autodiff Workshop*, Long Beach, CA, USA.
- Qin, H., Li, X., Liang, J., Peng, Y., and Zhang, C. 2016. Deepfish: accurate underwater live fish recognition with a deep architecture. *Neurocomputing*, 187: 49–58.
- Ren, S., He, K., Girshick, R., and Sun, J. 2015. Faster R-CNN: towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 28: 91–99.
- Ronneberger, O., Fischer, P., and Brox, T. 2015. U-Net: convolutional networks for biomedical image segmentation. *In International Conference on Medical Image Computing and Computer-assisted Intervention*, pp. 234–241. Springer.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z. *et al.* 2015. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115: 211–252.
- Saenko, K., Kulis, B., Fritz, M., and Darrell, T. 2010. Adapting visual category models to new domains. *In European Conference on Computer Vision*, pp. 213–226. Springer.
- Simonyan, K., and Zisserman, A. 2015. Very deep convolutional networks for large-scale image recognition. *In International Conference on Learning Representations*.
- Storbeck, F., and Daan, B. 2001. Fish species recognition using computer vision and a neural network. *Fisheries Research*, 51: 11–15.
- Strachan, N. J. C. 1993. Recognition of fish species by colour and shape. *Image and Vision Computing*, 11: 2–10.
- Sun, X., Shi, J., Dong, J., and Wang, X. 2016. Fish recognition from low-resolution underwater images. *In 2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pp. 471–476. IEEE.
- Tayama, I., Shimdate, M., Kubuta, N., and Nomura, Y. 1982. Application of optical sensor for fish sorting. *Refrigeration*, 57: 1146–1150.
- van der Walt, S., Schönberger, J. L., Nunez-Iglesias, J., Boulogne, F., Warner, J. D., Yager, N., Gouillart, E. *et al.* 2014. scikit-image: image processing in Python. *PeerJ*, 2: e453.
- Wang, D., and Shang, Y. 2014. A new active labeling method for deep learning. *In 2014 International Joint Conference on Neural Networks (IJCNN)*, pp. 112–119. IEEE.
- White, D. J., White, C. J., Svellingen, C., and Strachan, N. C. J. 2006. Automated measurement of species and length of fish by computer vision. *Fisheries Research*, 80: 203–210.
- Xie, S., and Tu, Z. 2015. Holistically-nested edge detection. *In Proceedings of the IEEE International Conference on Computer Vision*, pp. 1395–1403.
- Yang, Z., Luo, T., Wang, D., Hu, Z., Gao, J., and Wang, L. 2018. Learning to navigate for fine-grained classification. *In Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 420–435.
- Zheng, Z., Guo, C., Zheng, X., Yu, Z., Wang, W., Zheng, H., Fu, M. *et al.* 2018. Fish recognition from a vessel camera using deep convolutional neural network and data augmentation. *In 2018 OCEANS-MTS/IEEE Kobe Techno-Oceans (OTO)*, pp. 1–5. IEEE.
- Zion, B. 2012. The use of computer vision technologies in aquaculture—a review. *Computers and Electronics in Agriculture*, 88: 125–132.

Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Original Article

Automatic segmentation of fish using deep learning with application to fish size measurement

Rafael Garcia ^{1,2*}, Ricard Prados², Josep Quintana³, Alexander Tempelaar³, Nuno Gracias¹, Shale Rosen⁴, Håvard Vågstøl⁵, and Kristoffer Løvall⁵

¹Computer Vision and Robotics Institute, University of Girona, Campus Montilivi, Edif. P4, ES17003, Girona, Spain

²Girona Vision Research SL, Science and Technology Park of the University of Girona, c/ Pic de Peguera 11, Edif. Giroemprèn, ES17003, Girona, Spain

³Coronis Computing SL, Science and Technology Park of the University of Girona, c/ Pic de Peguera 11, Edif. Giroemprèn, ES17003, Girona, Spain

⁴Institute of Marine Research, P.O. Box 1870 Nordnes, NO-5817 Bergen, Norway

⁵Scantrol Deep Vision, Sandviksboder 1C, NO-5035 Bergen, Norway

*Corresponding author: tel: + 34 676 511 024; e-mail: rafael.garcia@udg.edu.

Garcia, R., Prados, R., Quintana, J., Tempelaar, A., Gracias, N., Rosen, S., Vågstøl, H., and Løvall, K. Automatic segmentation of fish using deep learning with application to fish size measurement. – ICES Journal of Marine Science, 77: 1354–1366.

Received 10 May 2019; revised 11 July 2019; accepted 14 August 2019; advance access publication 22 October 2019.

One of the leading causes of overfishing is the catch of unwanted fish and marine life in commercial fishing gears. Echosounders are nowadays routinely used to detect fish schools and make qualitative estimates of the amount of fish and species present. However, the problem of estimating sizes using acoustic systems is still largely unsolved, with only a few attempts at real-time operation and only at demonstration level. This paper proposes a novel image-based method for individual fish detection, targeted at drastically reducing catches of undersized fish in commercial trawling. The proposal is based on the processing of stereo images acquired by the Deep Vision imaging system, directly placed in the trawl. The images are pre-processed to correct for nonlinearities of the camera response. Then, a Mask R-CNN architecture is used to localize and segment each individual fish in the images. This segmentation is subsequently refined using local gradients to obtain an accurate estimate of the boundary of every fish. Testing was conducted with two representative datasets, containing in excess of 2600 manually annotated individual fish, and acquired using distinct artificial illumination setups. A distinctive advantage of this proposal is the ability to successfully deal with cluttered images containing overlapping fish.

Keywords: deep learning, fish sizing, trawl camera system

Introduction

According to the UN Food and Agriculture Organization, 33% of commercially important marine fish stocks worldwide are overfished (FAO, 2018). One of the causes of overfishing is that, in addition to targeted species, the fishing gears often catch other unwanted fish and marine life. Globally, nearly 11% of total catches are discarded because they are not the proper species or sizes (Pérez Roda *et al.*, 2019). In some cases, the quantity of this by-catch can exceed that of the targeted species. Excessive by-

catch is an immediate problem for fishers as it slows their catch sorting operations considerably, increases fuel consumption and wear on their fishing gear. Under management systems utilizing by-catch caps or closures to protect juveniles, fishing opportunities may be curtailed. In the long term, high levels of by-catch can contribute to overfishing jeopardize the long-term sustainability of the fishery.

Some countries and regions have enacted prohibitions on discarding unwanted catches. The most recent revision to the EU

Common Fisheries Policy (EU regulation 1380/2013) institutes a landing obligation requiring all catches of regulated commercial species to be landed and counted against quota. This includes catches of undersized individuals, which can be utilized to avoid waste, but not for direct human consumption or at a profit which could result in the establishment of markets.

Most fishermen use echosounders to detect fish schools and make qualitative estimates of the amount of fish and species present. Advanced “split beam” echosounders can give an indication of fish size, and characteristics such as frequency-response and school geometry can be used to differentiate between some species (Korneliusen *et al.*, 2009). However, systems to provide quantitative real-time species identification and measurement during fishing are largely in the demonstration phase (Pobitzer *et al.*, 2015; Berges *et al.*, 2018). As a result of this uncertainty, vessels relying on acoustics to target-specific species may catch undersized individuals or other species.

This paper proposes a novel fish sizing method when capturing fish using a trawl. The proposal is based on the use of the existing Deep Vision system (Rosen and Holst, 2013), directly placed in the trawl, to acquire stereo image pairs at a fixed frequency of five or ten images per second. The images are saved in a solid-state unit capable of storing ~ 1 million image pairs, equivalent to 60 h of data collection. In this paper, the images have been processed offline, but we aim at processing them onboard the Deep Vision system in the near future which will make real-time active sorting possible. This will enable more sustainable fishing activities by reducing catches of undersized individuals and unwanted species.

Material and methods

Data acquisition

Data were obtained on two testing cruises in the North Atlantic, the first in the North Sea onboard the Norwegian R/V “Dr Fridtjof Nansen” during March of 2017 (hereafter dataset 1), and the second in the Norwegian Sea with the chartered fishing vessel M/S “Vendla” during May of 2017 (hereafter dataset 2). Both vessels used an 832-m circumference pelagic trawl designed for surveys of small pelagic species in the Northeast Atlantic. Dataset 1 included images of saithe (*Pollachius virens*), blue whiting (*Micromesistius poutassou*), redfish (*Sebastes* spp.), Atlantic mackerel (*Scomber scombrus*), velvet belly lanternshark (*Etmopterus spinax*), and Norway pout (*Trisopterus esmarkii*), while dataset 2 included images of Atlantic mackerel, blue whiting, and Atlantic herring (*Clupea harengus*).

Acquisition of stereo image pairs of fish in the trawl was done using the Deep Vision system which is currently used to provide fisheries survey operations with information about depth and position of fish entering the sampling trawl. Using Deep Vision, it is also possible to conduct surveys which retain images rather than the actual fish. This lessens the environmental impact of the sampling and the workload of handling and measuring the catch. At the same time it provides images and metadata that can be used for length measurements and species classification. Combined with acoustic measurements this information provides higher confidence data used as input for stock assessment.

The Deep Vision system is divided into a subsea system and a topside system. The subsea system has a stereo camera, strobe lights, battery, and an enclosing studio frame designed for optimal image quality and consistency. The studio frame is integrated into the trawl to ensure smooth flow of catch through the system,

and protects the electronic components from the rigours of trawl handling and operations (see Figure 1).

The topside system provides a graphical user interface for size measurement and species classification, through a combination of manual and more automated processes. The output from the analysis software is combined with the data from the subsea system into an annotated dataset that can be used to produce statistical data.

During both surveys, the stereo image pairs were recorded at 5 fps, in JPG format, with an image resolution of 1392×1040 pixels. Lighting was provided by two synchronized strobes producing ~ 18000 lumen each at a colour of 4100 K. Although the lights were pointed to the ceiling and floor of the studio frame to provide diffused lighting, their angle varied slightly between cruises resulting in slight differences in reflection and illuminance inside the volume where objects pass through the Deep Vision canal (Figure 4). In addition, the user was allowed to make changes to camera exposure time, gain and gamma correction, introducing an additional source of inconsistency in image appearance. The impact of this uneven appearance on further image analysis prompted a full mechanical redesign of the lights to a production model with both higher total light output and fixed angle (Figure 1).

All the acquired images are analysed using the processing pipeline illustrated in Figure 2. First, images are pre-processed to correct nonlinearities and non-uniform lighting effects. Next, we use a Mask R-CNN architecture to localize and segment every individual fish in the image. The obtained segmentation is then refined in the next step using the local gradient to estimate the boundary of every fish. Finally, the length of the fish is computed exploiting stereo information. The different processing phases are detailed below.

Image pre-processing

Image pre-processing aims at correcting non-uniform lighting to produce images with a similar contrast between the fish and the background regardless of the location of the fish in the image. To carry out this correction, we should first linearize the image (Prados *et al.*, 2017).

Linearization is a desirable pre-processing step since cameras provide RGB values that are non-proportional with the incoming light energy. This is so because the human visual system has a nonlinear response (Burton, 1973). If an image encodes light in a $[0,255]$ interval, a value of 128 is perceived as half the lightness by the human eye, but in reality that point is reflecting (\sim) 25% of the light. That is, the camera response functions for all the colour channels are adapted to the human eye, and therefore they are nonlinear, especially if images have been stored using the JPG format, as it is often the case to minimize disc space to store large datasets. Therefore, since most processing algorithms assume that the value of a pixel is proportional to the amount of light collected by that specific pixel, linearizing the image would provide a better-conditioned set of pixel values for further processing. Moreover, using linearized images ensures providing the processing algorithms with a more accurate representation of the measured spectra, and consequently its behaviour and outputs become more consistent. In our case, images are linearized using the camera linearization method described in Debevec and Malik (1997). After this process, the RGB values become proportional with the irradiance on the sensor pixels, and the image is ready to

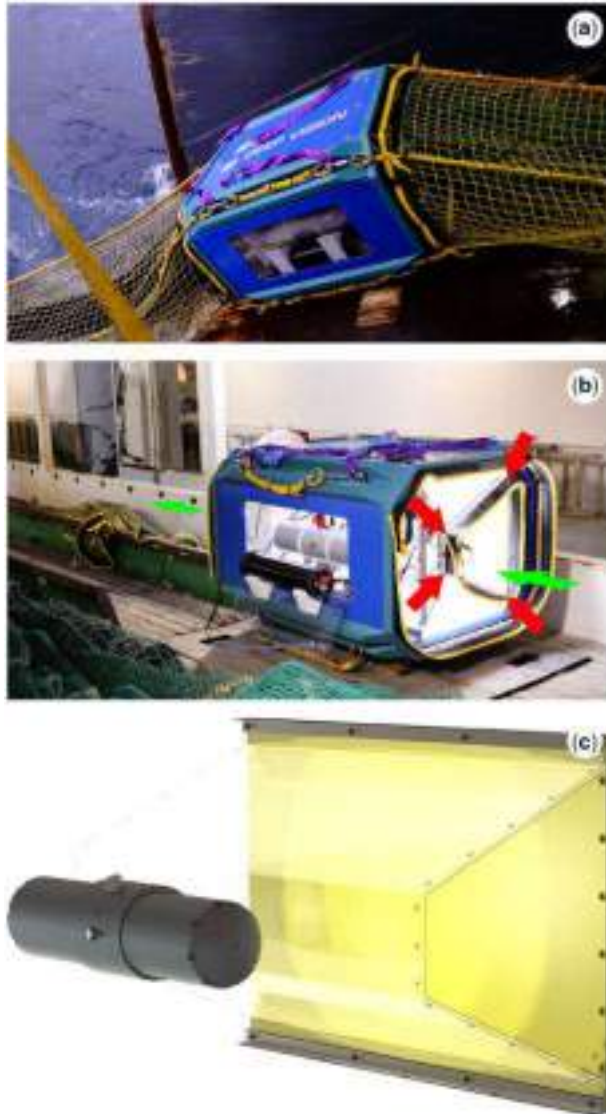


Figure 1. Deep Vision subsea system. The system is placed inside a trawl net (a) and contains a stereovision camera set and indirect lighting source. The arrows in the middle figure (b) define the “studio” section, corresponding to the area where the catch flows, and which can be seen in detail in the bottom schematic (c). Fish cross through a trapezoidal plexiglass section which ensures they maintain at least 20 cm distance from the cameras and lights and are within the field of view of the cameras.

undergo further linear operations, such as the correction of the non-uniform lightning. All subsequent operations are performed in linear RGB values.

Although the Deep Vision system provides images with a good overall illumination, the amount of light on the central area of the images is higher than that at the corners of the image. Therefore, once the images are linearized, we also correct the images for non-uniform lighting. To do this, we first convert the images from RGB to HSV (Hue, Saturation, Value), where V corresponds to the image luminance (Schwarz *et al.*, 1987). The luminance channel is the only component that will be used to correct the illumination effect. The illumination correction is performed by modelling the background, i.e. we compute the

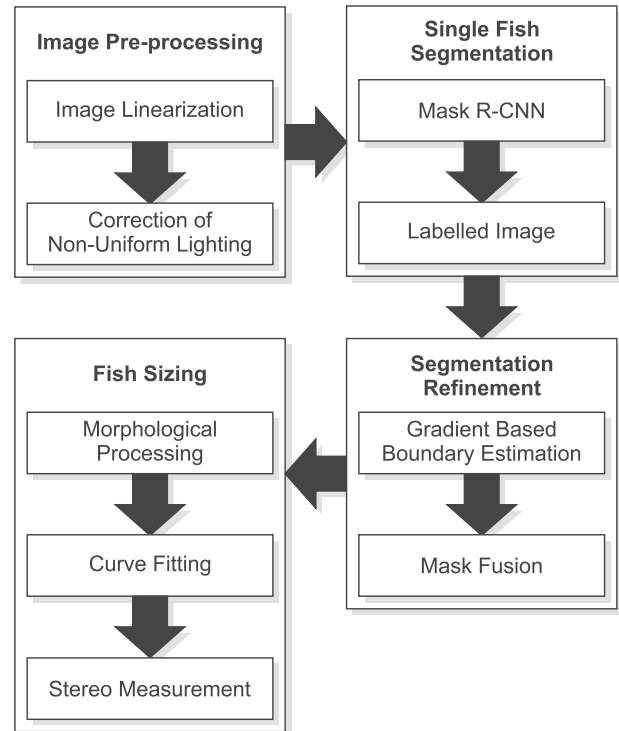


Figure 2. Automatic fish measurement pipeline. The process starts with the pre-processing of the image, and then a CNN localizes every individual fish. The CNN also provides a segmentation mask for the fish. Next, these masks are refined using local contrast information to delineate the boundary of every fish, and finally the length of the specimen is measured based on stereo cues.

median of a sufficiently large set of images of the scene (typically 300). The high power of the lighting system makes any external lighting contribution negligible, and consequently the illumination can be assumed as constant during the whole trawl. Ideally the images are selected at the beginning of the trawl haul before fish begin entering the field of view, although the only requirement is that, for the volume of 300 images, every pixel coordinate should not contain fish in slightly more than half the images (>150). The median value for each image pixel will be later on computed. If a given coordinate show no fish most of the time, the appropriate background value will be kept for this pixel location by the median measure. Once the median image has been computed from the V component of the set of images, we obtain a background luminance image that allows us to infer the illumination of the scene. The estimated background image is then inverted and applied as a non-uniform illumination compensation pattern to correct the luminance (V) of every image of the sequence. The RGB values of the final images are recovered from the HSV representation, ensuring that the correlation between the RGB channels is preserved, i.e. the original colours are kept.

It should be noted that working directly on the RGB colour space using channel-wise processing, as is commonly done in several image processing algorithms, may lead to a loss of the correlation between the values of the RGB triplets, thus shifting the original colours acquired by the camera.

Compensating the non-uniform illumination on all the images has proved to better condition the data to perform the subsequent fish segmentation (Prados *et al.*, 2014).

Single fish detection

Our aim is to be able to segment individual fish in the images, so that measuring the fish once it has been segmented becomes a trivial task. Figure 3 illustrates the problem we want to solve. Figure 3a shows a situation in which fish segmentation is quite easy since the background of the Deep Vision system can be modelled, and everything that is not background could be assumed to be a fish. However, Figure 3b shows a more challenging situation in which the fish to be measured are overlapping, making it difficult to determine their outline. In these situations in which we are not able to formalize an algorithm to recognize an object (e.g. a fish), using of machine learning methods has shown to be the best alternative. Among machine learning, deep convolutional neural networks (CNNs) have proved to be capable of achieving the best results on challenging datasets using supervised learning (Krizhevsky *et al.*, 2017). CNNs have also demonstrated good accuracy in automatic classification of species using simulated Deep Vision images (Allken *et al.*, 2019).

One of the state-of-the-art CNN-based deep learning object detection approaches is *Region-CNN* (or *R-CNN*). *R-CNN* provides a solution to the fast detection of regions of interest (RoI) within an image. Based on this approach, more complex architectures have recently appeared such as *Faster R-CNN* (Girshick, 2015) for faster speed object detection, as well as *Mask R-CNN* (He *et al.*, 2017) for object segmentation. In this paper, we use a *Mask R-CNN* architecture for fish detection and segmentation. *Mask R-CNN* combines *Faster R-CNN* for object detection in which the number of objects may vary from image to image, and fully convolutional networks (FCNs) for segmentation to establish what pixels in the image belong to what object. This step of detecting and delineating the boundaries of every individual object in an image is called “semantic segmentation,” and allows us to differentiate individual fish when two or more instances of a fish overlap in the image, as illustrated in Figure 3b.

Faster R-CNN performs individual fish detection in two stages. First, it determines the bounding boxes (i.e. RoIs) using the region proposal network (RPN) standard. The RPN is basically a lightweight neural network that scans the image in a sliding-window fashion to find regions that contain objects. Second, for each RoI it determines the class label of the object through RoI pooling. Therefore, *Mask R-CNN* incorporates these two stages, but it performs RoI pooling in such a way that there is no loss in stride quantization due to rounding when pooling is performed, as opposed to the rounding performed by *Faster R-CNN* (Ren *et al.*, 2015). Moreover, the sliding window is handled by the convolutional nature of the RPN, which allows it to scan all regions in parallel exploiting the GPU architecture.

FCNs are used to predict the mask for every RoI. Convolutional layers retain spatial orientation and this information is crucial for location-specific tasks such as creating a mask for every individual fish (He *et al.*, 2017). This is a clear advantage with respect to fully connected layers, in which the spatial orientation of pixels with respect to each other is lost as they are squeezed together to form a feature vector (Long *et al.*, 2015).

Our *Mask R-CNN* architecture was initially pre-trained for the COCO dataset (Lin *et al.*, 2014). Then, the last layer was modified to classify between fish and background and we re-trained the last layers using our fish training data for 20 iterations. This fine-tuning strategy allows us to reduce the training time and the

needed amount of data compared to training from scratch. Next, the full network was trained with our trawling data. In all cases, during training we tried to reduce overfitting on image data by artificially enlarging the dataset using data augmentation, which included image translations, horizontal and vertical reflections, rotations, and shear transformations.

Segmentation refinement

The mask computed by *Mask R-CNN* has been obtained using a low-resolution image. Thus, the mask that segments the fish has a lower accuracy than those that can be obtained from the full-resolution original images. Therefore, a final stage of mask refinement is applied to obtain a much finer spatial layout of the fish, i.e. a more accurate segmentation.

The blobs estimated by *Mask R-CNN* are first scaled and transferred to the full-resolution image (1228 × 1027 pixels). Then, the gradients of the *V* channel on the original image are computed. This results in an image where the boundary of the objects is clearly distinguishable. The gradient magnitudes are thresholded to keep only the higher values, that is, the most prominent boundaries. Finally, both the *Mask R-CNN* masks, resulting in most cases in conservative segmentation, and the gradient-based boundary refinement masks, are fused into a single one for each image object. Empty inner areas are filled using binary morphological operators.

In case of overlapping fish, *Mask R-CNN* masks are used to guess where the boundaries of every specimen should be placed, given that the gradient-based refinement cannot distinguish among different objects. To determine which pixel belongs to each fish, *Mask R-CNN* masks are dilated using a customized multi-label dilate operation, which stops growing in a given direction when another neighbouring object is growing in the opposite direction and colliding with the first. The result of this dilate operation is used to determine the contribution of the gradients image to each fish mask.

Segmentation performance

To evaluate the performance of the masks obtained by our processing pipeline, a detection accuracy measure is required. A standard set of metrics [intersection over union (IoU) and pixel accuracy] is used to quantify the segmentation results, since they are the *de facto* evaluation metrics used in object detection. IoU, also referred as Jaccard index, is an evaluation metric used to measure the accuracy of object segmentation on a particular dataset. IoU is often computed using the bounding box predicted by the CNN detector and the ground-truth (i.e. hand labelled) bounding box. In our case, since our detector generates a pixel region (mask) containing the pixels that correspond to a given fish, and the ground-truth is also a hand-labelled pixel region, IoU is computed using these two regions. The final score is obtained by dividing the area of overlap of the predicted region and the ground-truth region by the area of union of both the predicted region and the ground-truth region:

$$\text{IoU} = \frac{\text{ground-truth} \cap \text{prediction}}{\text{ground-truth} \cup \text{prediction}}$$

However, the measure of pixel accuracy corresponds to the percentage of pixels in the image which were correctly classified.

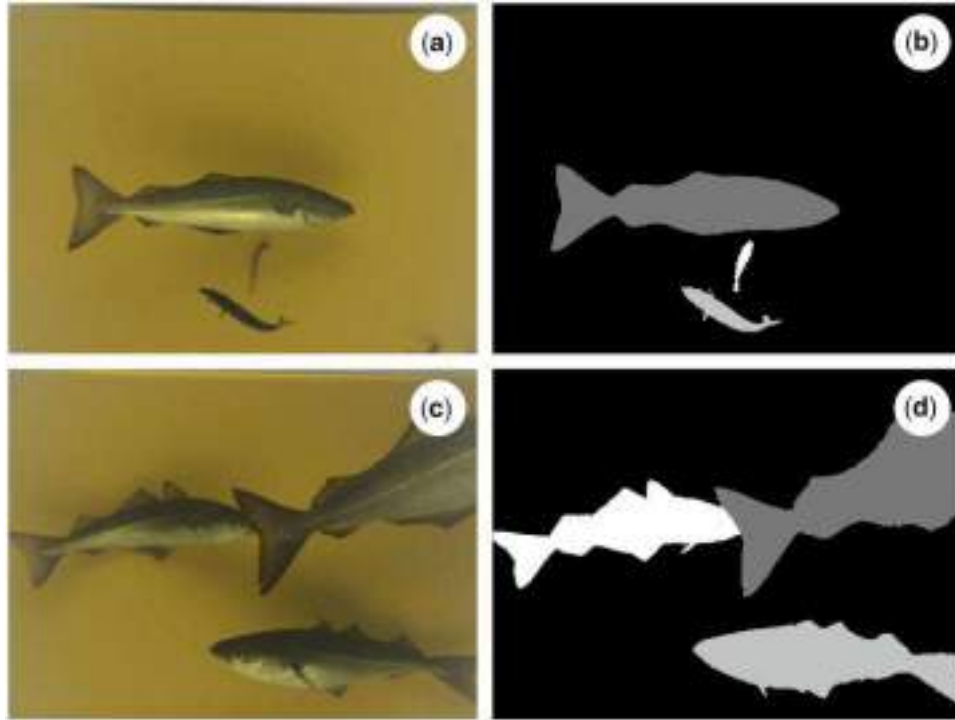


Figure 3. Fish segmentation. In simple cases such as (a), fish can be segmented into individual specimens simply by background subtraction (b). However, we need a cognitive understanding of the image to be able to segment the three fish instances in (c) shown in (d).

Usually it is presented for each class and the mean of all classes is provided. In our case both values are the same as we only have the “fish” class.

For this metric we need to introduce the notions of TP, TN, FP, and FN. True positive (TP) represents a pixel that is correctly predicted to belong to the given class whereas a true negative (TN) represents a pixel that is correctly identified as not belonging to the given class. False positives (FP) and false negatives (FN) are defined accordingly. The accuracy metric is then computed as

$$\text{accuracy} = \sum \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}.$$

Length estimation

Once the specimens have been properly segmented, the final stage consists of finding a line that accurately describes the length of the fish. For this purpose, we estimate the fish skeleton using morphological operations applied to the labelled image, but it should be noted that the actual length of the fish should be estimated taking into account its 3D pose. The thinning morphological operation involves eroding the segmented region until skeleton level (Dougherty, 1992), i.e. shrinking the region corresponding to the individual fish until the blob becomes 1 pixel wide. This typically leads to a line centred along the main axis of the fish. Before performing morphological skeletonization, the binary masks resulting from the segmentation of the previous section are smoothed by applying a “closing” morphological operation. In this way, a continuous and typically smooth line is obtained, representing the main axis of the fish.

The next step is the estimation of a curve following the trajectory described by the pixels of the skeleton. Once the points

defining the skeleton have been obtained, a cubic polynomial is estimated using RANSAC (Fischler and Bolles, 1981). In this way, the points of the skeleton are classified in inliers and outliers, and after a number of iterations, a consensus solution is computed by least squares fit of the largest set of inliers, obtaining the final estimation of the curve.

Once the curve equation is derived, the starting and ending points defining the length of the fish are determined as the intersection between the estimated curve and the boundaries of the smoothed fish blob. Since the stereo system has been calibrated and the images rectified (Hartley and Zisserman, 2003), these points can then be easily transferred from the right to the left image of the stereo pair by applying the axis constraints determined by the stereo rectification. Then, once front and back points have been established in both images of the stereo pair, a set of uniformly distributed points along the curve are selected in the right image. These points are transferred to the left image following the same uniform distribution, using the image rectification to determine its Y location. Finally, the set of measurement point pairs from the right and left images is used to compute the distances of the segments connecting them using epipolar geometry, thanks to the calibration of the stereo system.

Results

A total of 1805 manually annotated images (corresponding to the left camera of the stereo pairs) have been used to validate the pipeline proposed in this paper, with a total of 2629 fish annotations. These images have been acquired in two different cruises. Dataset 1, including 1605 annotated images, was acquired by R/V Dr Fridtjof Nansen on March 2017. This dataset represents a small subset of all the images acquired during the survey, and includes frames from three different hauls (138 055 stereo pairs).

Dataset 2 was acquired by F/V Vendla on May 2017 and it includes 200 annotated images, all of them from the same haul (28 117 stereo pairs). Both surveys consist of thousands of images, but only small samples containing fish suitable for an appropriate labelling (a large percentage of images contain no fish at all) can be used. The annotation effort is significant, taking into account that the labelling procedure implies a precise manual segmentation of each specimen, not a simpler approximate bounding box specification.

Figure 4 illustrates the appearance of the images of both datasets, as well as the result of correcting non-uniform illumination. It should be noted that the appearance of the images in both datasets is different due to the change of lighting arrangement and camera parameters (with a gain factor of 1.2 in case of dataset 1 and gain factor of 2 in case of dataset 2). In dataset 2, the central part of the image is considerably brighter than in dataset 1, and as a consequence, the margins of the image are darker than in the first dataset. After applying the strategy to compensate the non-uniform lighting, using a specific per-haul pattern to maximize precision, the images of both datasets become better suited for posterior processing. The frames attain a more even appearance, with uniform light distribution, making the contained data better conditioned for the subsequent steps.

Two different sets of experiments have been conducted. In the first experiment, we aimed at evaluating the capability of the architecture to generalize the problem of fish detection by training using the 1605 images of dataset 1, and then testing on the 200 annotated images of dataset 2, in which lighting conditions and camera settings have changed.

It should be noted that the two datasets also present different characteristics in terms of the type of fish present. Saithe dominated in the first cruise, which also included blue whiting, redfish, Atlantic mackerel, velvet belly lanternshark, and Norway pout. The second cruise included images of Atlantic mackerel, blue whiting, and Atlantic herring. In addition to these fish, the second dataset also included northern krill, *Meganyctiphanes norvegica*, in most images. Moreover, the average number of fish per image is also much larger in the second dataset.

The Mask R-CNN was trained with the images of dataset 1, acquired by the R/V “Dr Fridtjof Nansen,” but applying the data augmentation techniques described above. The original dataset was split into 80% for training and 20% for validating.

After finishing this training we applied the obtained weights on 200 annotated images from the second dataset acquired by F/V “Vendla.” This dataset is completely independent from the images used for training and validation. Test images were previously segmented by hand, creating a ground-truth to compare all methods. Fifty of these images contain overlapping fish while the other 150 contain one or more fish, but with no overlap. Table 1 illustrates the results obtained in this first trial.

Analysing the values of Table 1, the reader would think that the CNN is doing a good job. We differentiate between “single fish,” which is the detection of fish when the masks corresponding to the fish are not connected to each other (see Figure 3a), and “overlapping fish,” which corresponds to the cases in which these masks overlap (see Figure 3b, central fish). In Table 1, IoU is ranging between 0.84 for “single fish” detection, and 0.82 for “overlapping fish.” And the accuracy is even higher with values of >0.98 in both cases. Therefore, at first glance, the Mask R-CNN architecture seems to have done a good job to generalize the problem of fish detection.

It should be noted, however, that in our case we want to segment every isolated fish to enable its later sizing. In the case of overlapping fish (see Figure 3b), applying IoU out of the box would only take into account if a pixel that was predicted as class “fish” belongs to a fish in the ground-truth. However, this is not what we need in our application. Consider the example of Figure 5. The ideal ground-truth masks are shown in Figure 5a, with the red fish labelled as 1 and the blue fish with label 2. Figure 5c shows a fish segmentation in which the two overlapping fish are detected as a single fish. This would be considered as a very good segmentation in the standard IoU metric frequently used in the literature, e.g. (He *et al.*, 2017), but in our case we consider this a bad result since it is missing the detection of fish 2, and over-segmenting fish 1. Therefore, we introduce a new metric, namely IoU*, to measure IoU on a slightly different way that better serves our purposes. This measurement of IoU* will work as follows. An IoU* measurement will be computed for every fish in the ground-truth. The IoU* corresponding to the red fish as the area of intersection between the red region in Figure 5a and the red area in Figure 5c, and that value will be divided by the union of the same two regions. In this way, the detection of fish 1 will have a low IoU, as we will divide by a large area of union. Equally, for fish 2 we will divide the area of intersection by the total area of union of Figure 5b plus the blue area of Figure 5a, also producing a low IoU* value since it will have also a large number in the denominator. Using this metric, large values of IoU* guarantee that only one fish has been detected, while low values indicate that two or more overlapping fish in the ground-truth have been predicted as a single fish in the detection phase. Experimentally, this threshold has been set as 0.7.

The results of this new metric are given in Table 2. Again, we distinguish between the previous two cases depending on whether fish are overlapping to have a better insight of the performance of the system under this critical situation. In the first two columns the table details the number of images of the second dataset, and the total number of fish manually annotated in those images. The third column states how many of these fish are detected with an IoU* with a value of >0.7 , which intuitively means that the detection is good, i.e. two fish in the ground-truth are detected as two fish in the trial, and not as a single, larger fish. For the case of single fish (non-overlapping) we observe that 334 fish are correctly detected out of the 368 fish in the ground-truth. This is really a good performance if we take into account that several of the fish manually annotated in the datasets correspond to partially visible fish that are entering or leaving the field of view of the camera. However, for the images in which fish are overlapping, only 154 out of 272 fish are detected with an IoU* >0.7 . And 94 fish are detected with IoU* <0.7 , i.e. one fish is detected when >1 fish appeared in the ground-truth. It can be observed that, as opposed to what it seemed in Table 1 using the standard IoU metric, the performance of Mask R-CNN in this first trial is not so great, especially in the case of overlapping fish. The next two columns present the number of false negatives, i.e. fish not detected at all, and false positive. In this dataset the false positives normally correspond to the prediction of fish in areas of the image that correspond to northern krill, present in all the images of sequence 2. Finally, the last column corresponds to the average IoU* measurement, giving a value of 0.76 for the single fish case, and 0.58 in the case of overlapping fish. It should be noted that this average is computed from all the IoU* values of all the

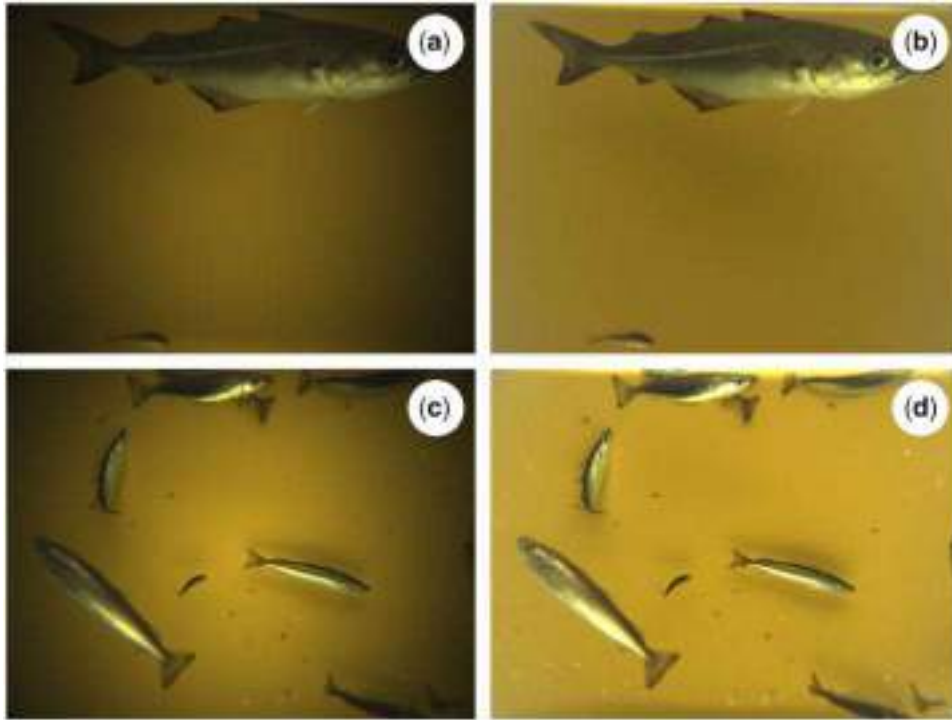


Figure 4. Correction of non-uniform illumination in dataset 1 (top) and dataset 2 (bottom). (a) Image from the Dr Fridtjof Nansen March 2017 dataset. (b) Image after non-uniform illumination compensation. (c) Image corresponding to the Vendla May 2017 dataset. Note the different appearance of the image with respect to (a). The centre of the image is brighter, while the boundary areas are still significantly dark. (d) Image after non-uniform illumination compensation.

Table 1. Results obtained by Mask R-CNN.

	IoU	Accuracy
Single fish	0.845	0.994
Overlapping fish	0.824	0.984

The network was trained using dataset 1, and the test has been quantified using the images of dataset 2. The results suggest a very good generalization capability of the network for detecting fish.

images in the corresponding dataset. We average all IoU* values for every fish in the ground-truth, but we also accumulate and account for 0 if FN or FP occur in the test images. Therefore, our average IoU* metric strongly penalizes false detections.

The last two rows of Table 2 detail the results of taking the fish detection masks obtained in this first trial by Mask R-CNN and applying the gradient mask refinement to them. We notice that gradient refinement is not able to improve fish detection, although it raises IoU* to 0.80 and 0.61, respectively. This basically means that the segmentation mask is more accurate after gradient refinement.

Table 3 reports the results of the second experiment. In this case, both datasets were used to create the train, validation, and test sets. Out of the total number of images (1805), roughly a 10% is used to evaluate the final model fit on the training dataset (test set), and the remaining 90% of the images were further divided into 80% for training and 20% for validation to tune the hyperparameters of the Mask R-CNN. Again, to better understand the performance of the network, we divided the test set images between (a) single fish and (b) overlapping fish situations.

For the single fish scenario, as expected, we see that the performance of the detection is better than in the first experiment, since the training data includes images of datasets 1 and 2. More than 96% of the fish are correctly detected when there is no overlapping fish, i.e. 225 correct detections from 233 annotated fish. This percentage goes down to roughly 79% when fish are occluded by other fish. These results with overlapping fish drastically improve the results of experiment 1, with 57% of correct detections of overlapping fish. It can also be observed that the number of FN and FP has also been drastically reduced with respect to the previous trial. Finally, the last column of Table 3 includes IoU* average values of 0.89 and 0.79 for non-overlapping and overlapping fish, respectively. These values are slightly improved by the gradient refinement technique, on 0.01 in every case. This is a sign that the masks generated by Mask R-CNN in the second experiment are more accurate than the ones predicted in the first trial, but can still be improved through gradient refinement. Some sample results of the second experiment can be shown in Figures 6 and 7.

Figure 7 shows intermediate qualitative results of the proposed pipeline. It can be observed how the individual fish segmentation algorithm provides a much better fish delineation with respect to the labelled image provided by Mask R-CNN.

Discussion and conclusions

Fish length estimation and catch composition are among the most crucial information collected in fisheries research. The Deep Vision system allows fishing vessels to collect stereo imagery, and proper processing of these data enables gaining critical information about average fish size and catch composition during the trawling operation.

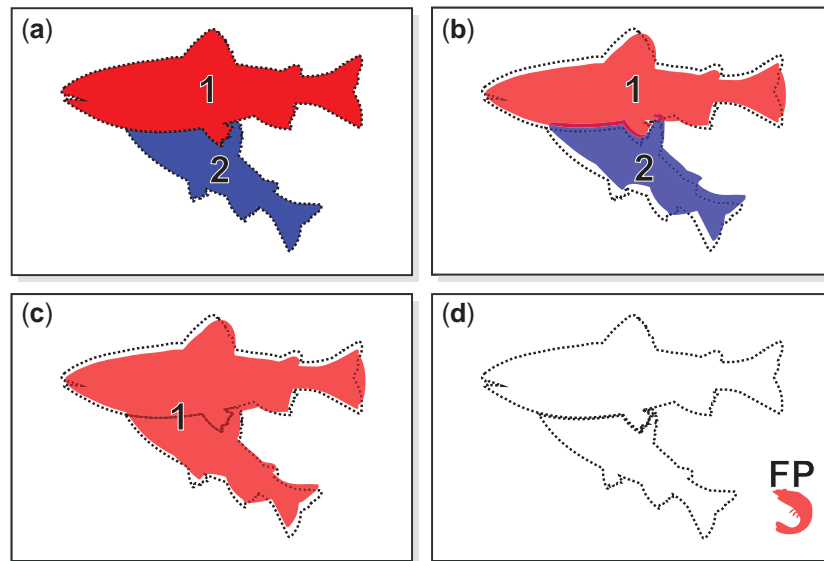


Figure 5. Fish masks. (a) Ground-truth hand annotation. (b) Example of masks detected by the CNN. The dashed lines show the corresponding ground-truth. The coloured area outside the dashed region corresponds to a false positive area, the white area inside the dashed region defines a false negative. (c) Example of an incorrect segmentation in which the CNN detects as a single instance the two fish of (a). (d) False detection of a non-existent fish, giving rise to another false positive.

Table 2. Experiment 1: results obtained by Mask R-CNN after training with dataset 1 (D#1) and testing with dataset 2 (D#2).

		No. of images	Total no. of annotated fish	No. of detected fish with $\text{IoU}^* > 0.7$	No. of detected fish with $\text{IoU}^* < 0.7$	FN	FP	IoU^*
Mask R-CNN train and valid. on D#1 + test on D#2	Single fish	150	368	334	15	19	25	0.76
	Overlapping fish	50	272	154	94	24	16	0.58
Gradient refinement	Single fish	150	368	333	16	19	24	0.80
	Overlapping fish	50	272	156	95	21	15	0.61

Performance taking into account the new metric IoU^* that penalizes detection of a single fish when two or more fish instances are labelled in the ground-truth.

Several works in the literature have tried to segment fish in underwater video sequences. Some achieve fish detection based on matrix decomposition (Qin *et al.*, 2014) or exploiting texture and shape features that characterize fish with respect to the background (Spampinato *et al.*, 2010). Other works rely on salient features (Fernandes *et al.*, 2016), carefully selected double thresholds (Chuang *et al.*, 2016), or the guided filter (Sanchez-Torres *et al.*, 2018). In many cases, the approach involves a static camera that allows modelling the background to then isolate the fish to carry out monocular detection or stereo measurements (Costa *et al.*, 2006; Pérez *et al.*, 2018), while other works train-specific Deep Learning architectures for fish classification (Qin *et al.*, 2016). However, in all cases the detected fish were not overlapping with other fish in the field of view of the camera. Proper delineation of individual fish in overlapping situations still remains a challenge.

Stereo imaging is often employed to obtain depth information, and depth cues can be used to segment RoI in some well-conditioned situations. However, traditional stereo matching techniques such as *Semi Global Matching* (Hirschmuller, 2005) or *Block Matching* (Konolige, 1998) fail to reliably detecting the fish boundaries in cluttered situations, as depicted in Figure 8. Depth cues from stereo alone can potentially be used to separate fish standing at clearly different distances, such as in the case of

Figure 8a and b. On the contrary, we find in our datasets many cases in which multiple fish stand at approximately the same distance while overlapping, or are imaged while being significantly rotated from the ideal fronto-parallel configuration (such as in Figure 8e and f). In these situations, stereo matching fails to provide enough information to successfully and robustly separate the fish (Figure 8g and h). Figure 9 illustrates the result of our approach for this particular complicated case. While the result is not perfect in Figure 9b, it can nonetheless be considered as a successful detection and separation.

The processing pipeline proposed in this paper is able to provide accurate segmentations of individual fish in images acquired during standard fisheries surveys using the Deep Vision commercially available system. The pipeline involves three main phases: pre-processing, CNN-based segmentation, and gradient refining. Each phase contributes decisively to the performance of the overall system.

Pre-processing aims at exploiting the fact the imaging acquisition setup is well defined and constrained in terms of optical sensors, illumination characteristics, and background. By performing adequate modelling of the camera response and background illumination field, the variability of the visual appearance is reduced across different datasets and surveys. This, in turn, promotes the performance of the CNN, and, to

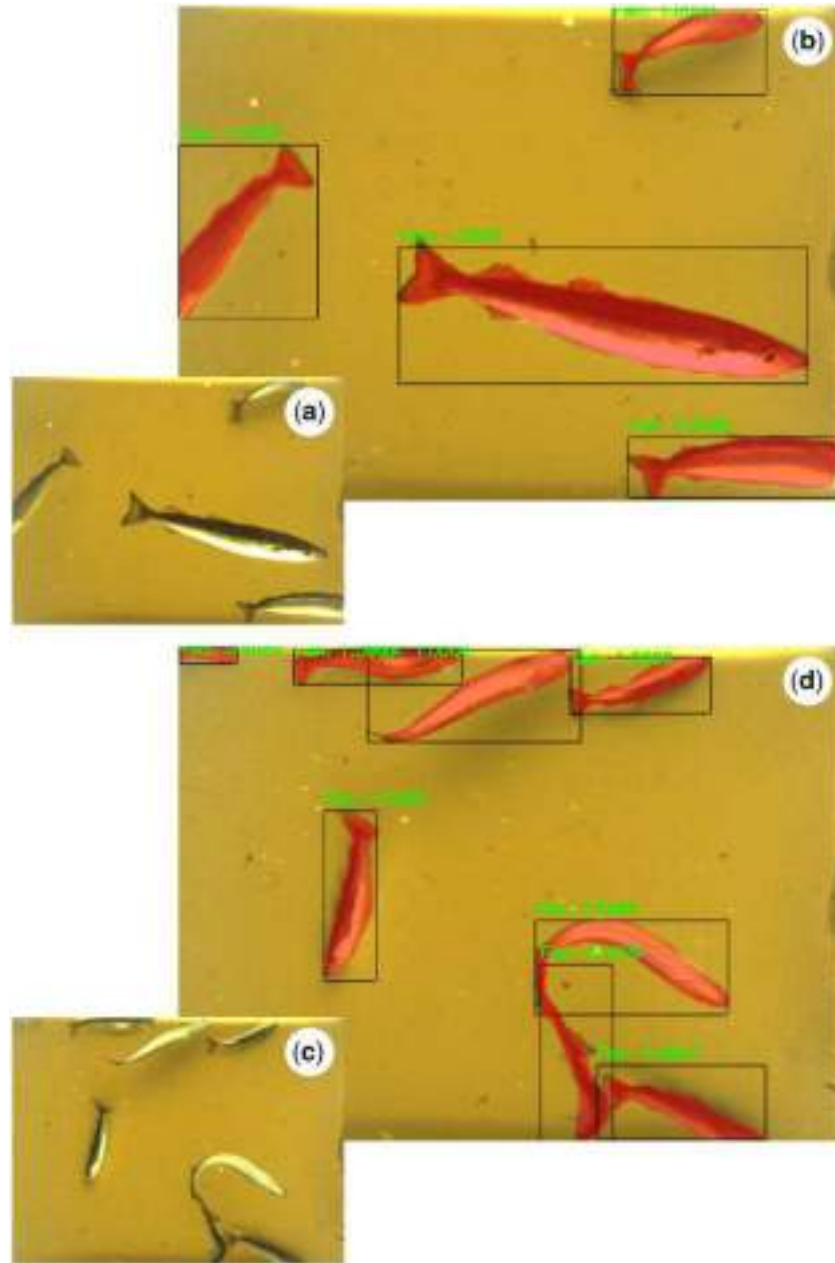


Figure 6. Fish detection and semantic segmentation performed by Mask R-CNN. (a) and (c) correspond to the original images. (b) and (d) illustrate the outcome of the algorithm. Note how Mask R-CNN is also able to detect overlapping fish, as shown in (d).

a lesser extent, also benefits the gradient refinement step at the end.

The Mask R-CNN architecture was selected for the *CNN-based segmentation*. A central reason behind this choice was its superior performance reported by He *et al.* (2017), when compared to closely related instance-aware alternatives such as Multi-task Network Cascades (Dai *et al.*, 2016) and Fully Convolutional Semantic Segmentation (Li *et al.*, 2017).

Finally, the gradient refining phase improves the delineation of the fish by using local contour cues. The impact of this step is clearly visible on Tables 2 and 3 regarding the IoU^* measurement, where there was a noticeable improvement. The

improved delineation is also of clear benefit for fish sizing accuracy.

In this study, we have also proved that standard IoU values are not adequate to quantify the performance of segmentation of individual fish in the overlapping situations in which specimens are occluded by other fish. A modification of the previous metric has been proposed (IoU^*) as a statistic that can effectively be used for gauging the similarity of the detected masks with respect to the hand-labelled ground-truth masks.

The approach in this paper has been developed with the operational goal of achieving real-time execution on dedicated hardware inside the Deep Vision imaging system. The testing

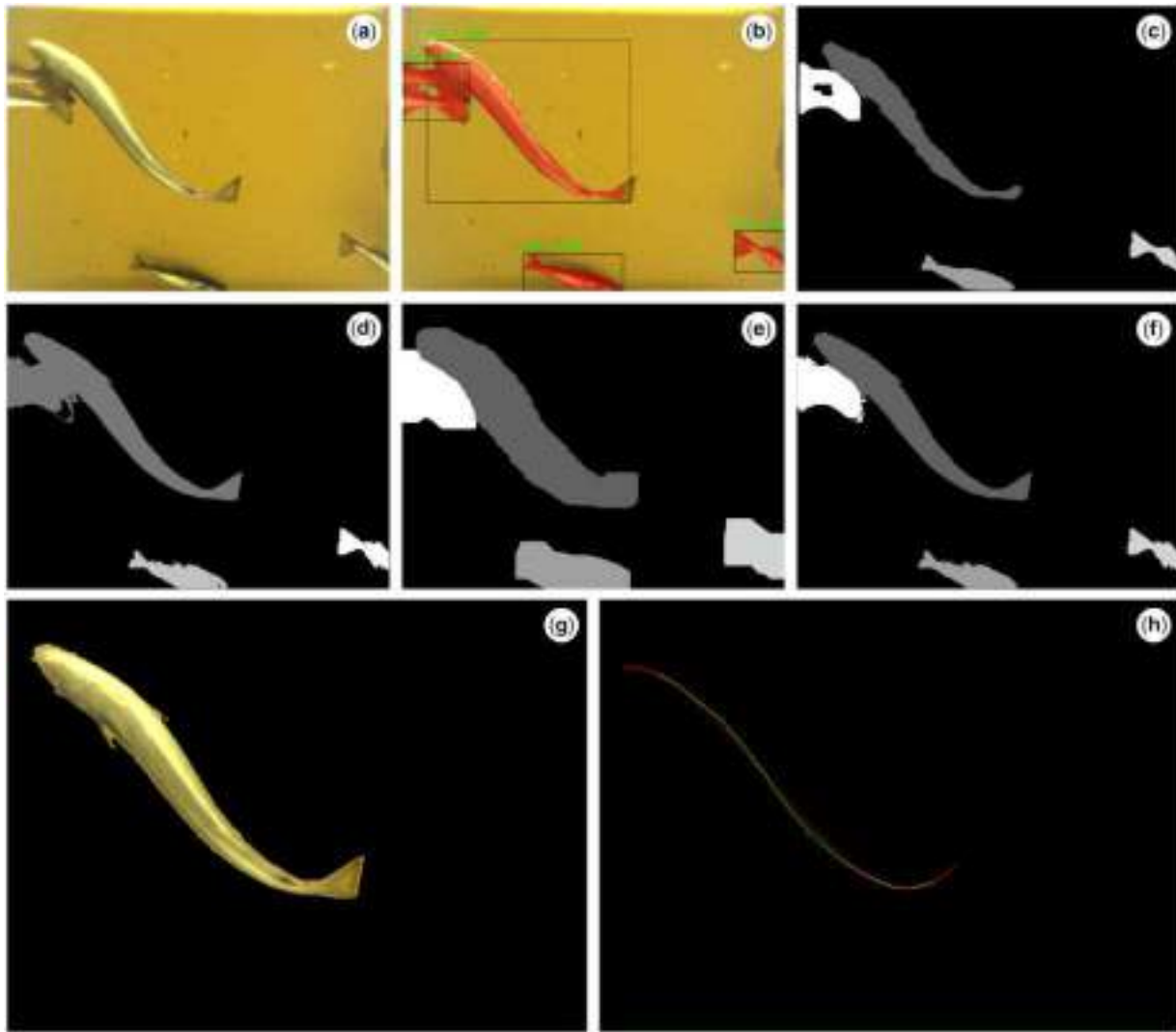


Figure 7. Automatic fish detection and length estimation. (a) Original image. (b) Fish detection and semantic segmentation through the Mask R-CNN processing. Note that the system is able to correctly detect the central fish, although it fails to detect the two tails on the left as two separate fish. (c) Labelled image as provided by Mask R-CNN. (d) Fish boundary gradient refinement mask. Note that, in this case, the segmentation is not able to distinguish among touching fish. (e) Multi-label dilate morphological operation of the Mask R-CNN segmentation. (f) Fish mask resulting of the combination of both gradient refinement and multi-label dilate. (g) Final segmented fish. (h) Skeleton pixels (in green) of the segmented fish and measurement points (in red) of the estimated fish-shape curve used to perform an automatic size measurement.

Table 3. Experiment 2: results obtained by Mask R-CNN after training with randomly selected 90% images from dataset 1 (D#1) and dataset 2 (D#2), the other 10% is reserved for testing.

		No. of images	Total no. of annotated fish	No. of detected fish with $IoU^* > 0.7$	No. of detected fish with $IoU^* < 0.7$	FN	FP	IoU^*
Mask R-CNN train and valid. on 90% (D#1 + D#2), test in 10% (D#1 + D#2)	Single fish	170	233	225	7	1	10	0.89
	Overlapping fish	26	104	82	16	6	5	0.79
Gradient refinement	Single fish	170	233	224	8	1	10	0.90
	Overlapping fish	26	104	84	14	6	4	0.80

Performance taking into account the new metric IoU^* that penalizes detection of a single fish when two or more fish instances are labelled in the ground-truth.

reported in this paper was conducted offline on a high-end desktop computer with a NVIDIA TITAN V GPU. The segmentation was run on the GPU at a frame rate was 2.67 images per second. The refinement in the current state is not optimized for speed.

A number of extensions to this work is planned in the near future. The validation of the size measurements is currently being pursuit with the intent of using fish specimens or accurate fish shape reproductions of known dimensions. The testing is to be conducted in water, to take into account the

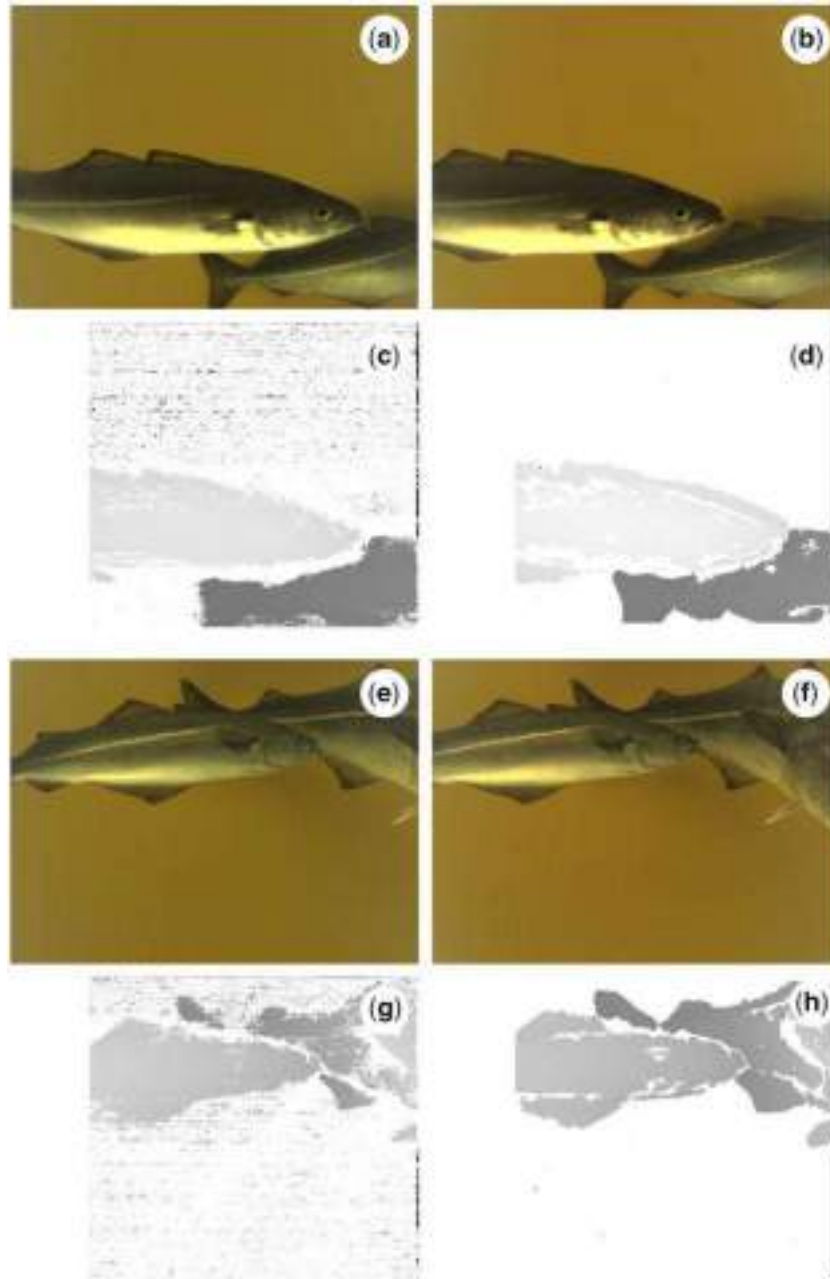


Figure 8. Traditional stereo matching techniques fail to segment overlapping fish due to lack of sufficient salient features and visual texture. The (a, b) and (e, f) images correspond to a pair of stereo images presenting fish that are partially occluded by other fish. (c, d) and (g, h) show the resulting disparity maps using two standard stereo processing techniques: (left) Semi Global Matching (Hirschmuller, 2005) and (right) Block Matching (Konolige, 1998).

refraction effects of the flat-port camera housing and how it affects the stereo geometry.

A second extension is directed towards achieving an execution frame rate in the order of 10 fps, on the target embedded processing hardware. This hardware is based on NVIDIA Jetson AGX Xavier modules and will be deployed with Deep Vision imaging system. The intended frame rate will allow performing tracking of fish across time, given that multiple instances of the same fish are likely to occur when images are acquired at 10 fps or higher, for nominal trawling speeds. This will enable the ability of estimating in real time the amount of fish in the trawl as well as the average

size. Finally, as more data becomes annotated, future development will extend this work to use Mask R-CNN for automatic fish species identification.

Funding

Development of Deep Vision technology has been supported through the Research Council of Norway's Industrial PhD Programme and Innovation Norway's program for development of environmental technology (project 100424). R. Garcia and N. Gracias were partly funded by the Spanish Ministry of Education, Culture, and Sport under project CTM2017-83075-R. Data

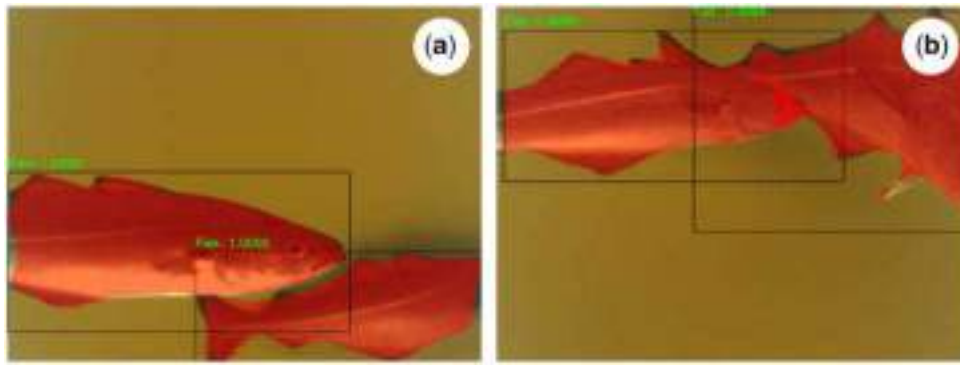


Figure 9. (a) Result of Mask R-CNN for the images of Figure 8b. (b) Instance segmentation of the two fish of Figure 8f: note the small error in the detection of the fish on the right. Although far from achieving ideal results, Mask R-CNN outperforms the state-of-the-art stereo processing techniques of Figure 8, even when the fish are not completely visible.

collection onboard R/V “Dr Fridtjof Nansen” was supported by the Institute of Marine Research under the CRISP centre for research innovation (Research Council of Norway project 203477) and vessel time onboard M/S “Vendla” was provided by the REDUS project with funding from the Norwegian Ministry of Trade, Industry, and Fisheries. The authors would like to thank Roger Portas for his assistance with this project.

References

- Allken, V., Olav, N., Rosen, S., Schreyeck, T., Mahiout, T., and Malde, K. 2019. Fish species identification using a convolutional neural network trained on synthetic data. *ICES Journal of Marine Science*, 76: 342–349.
- Berges, B., Sakinan, S., and van Helmond, E. 2018. Practical Implementation of Real-time Fish Classification from Acoustic Broadband Echo Sounder Data—RealFishEcho Progress Report. Wageningen Marine Research (University & Research Centre), Wageningen. Wageningen Marine Research Report, C062/18. 42 pp.
- Burton, G. J. 1973. Evidence for non-linear response processes in the human visual system from measurements on the thresholds of spatial beat frequencies. *Vision Research*, 13: 1211–1225.
- Chuang, M., Hwang, J., and Williams, K. 2016. Automatic fish segmentation and recognition for trawl-based cameras. *In Computer Vision and Pattern Recognition in Environmental Informatics*, pp. 79–106. Ed. by J. Zhou, X. Bai, and T. Caelli. IGI Global, Hershey, PA.
- Costa, C., Loy, A., Cataudella, S., Davis, D., and Scardi, M. 2006. Extracting fish size using dual underwater cameras. *Agricultural Engineering*, 35: 218–227.
- Dai, J., He, K., and Sun, J. 2016. Instance-aware semantic segmentation via multi-task network cascades. *In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3150–3158. DOI: 10.1109/CVPR.2016.343.
- Debevec, P. E., and Malik, J. 1997. Recovering high dynamic range radiance maps from photographs. *In Proceedings of the 24th annual conference on Computer graphics and interactive techniques (SIGGRAPH '97)*. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, pp. 369–378.
- Dougherty, E. 1992. *An Introduction to Morphological Image Processing*. SPIE Optical Engineering Press. ISBN0-8194-0845-X.
- FAO. 2018. *The State of World Fisheries and Aquaculture. Meeting the Sustainable Development Goals*, Rome, Italy. <http://www.fao.org/3/i9540en/I9540EN.pdf>.
- Fernandes, P. G., Copland, G., Garcia, R., Nicosevici, T., and Scoulding, B. 2016. Additional evidence for fisheries acoustics: small cameras and angling gear provide tilt angle distributions and other relevant data for mackerel surveys. *ICES Journal of Marine Science*, 73: 8.
- Fischler, M. A., and Bolles, R. C. 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24: 381–395.
- Girshick, R. 2015. Fast R-CNN. *In IEEE International Conference on Computer Vision (ICCV)*, Santiago, 2015, pp. 1440–1448. DOI: 10.1109/ICCV.2015.169.
- Hartley, R., and Zisserman, A. 2003. *Multiple View Geometry in Computer Vision*. 2nd edn, Cambridge University Press, New York, NY.
- He, K., Gkioxari, G., Dollár, P., and Girshick, R. 2017. Mask R-CNN. *In IEEE International Conference on Computer Vision (ICCV)*, Venice, pp. 2980–2988. DOI: 10.1109/ICCV.2017.322.
- Hirschmuller, H. 2005. Accurate and efficient stereo processing by semi-global matching and mutual information. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 807–814. IEEE, San Diego, CA.
- Konolige, K. 1998. Small vision systems: hardware and implementation. *In Proceedings of the 8th International Symposium in Robotic Research*, Springer, London, pp. 203–212.
- Korneliussen, R. J., Heggelund, Y., Eliassen, I. K., and Johansen, G. O. 2009. Acoustic species identification of schooling fish. *ICES Journal of Marine Science*, 66: 1111–1118.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. 2017. ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60: 84–90.
- Li, Y., Qi, H., Dai, J., Ji, X., and Wei, Y. 2017. Fully convolutional instance-aware semantic segmentation. *In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2359–2367.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P. *et al.* 2014. Microsoft COCO: common objects in context. *In European Conference on Computer Vision (ECCV)*, Springer International Publishing, pp. 740–755. DOI: 10.1007/978-3-319-10602-1.
- Long, J., Shelhamer, E., and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. *In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Boston, MA, 2015, pp. 3431–3440. DOI: 10.1109/CVPR.2015.7298965.
- Pérez, D., Ferrero, F. J., Alvarez, I., Valledor, M., and Campo, J. C. 2018. Automatic measurement of fish size using stereo vision. *In IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, pp. 1–6.
- Pérez Roda, M. A. (ed.), Gilman, E., Huntington, T., Kennelly, S. J., Suuronen, P., Chaloupka, M., and Medley, P. 2019. A third


- assessment of global marine fisheries discards. FAO Fisheries and Aquaculture Technical Paper, 633. FAO, Rome. 78 pp.
- Pobitzer, A., Ona, E., Macaulay, G., Korneliussen, R., Totland, A., Heggelund, Y., and Eliassen, I. K. 2015. Pre-catch sizing of herring and mackerel using broadband acoustics. *In* ICES Symposium on “Marine Ecosystem Acoustics (Some Acoustics)—Observing the Ocean Interior in Support of Integrated Management”, pp. 25–28. Nantes, France.
- Prados, R., Garcia, R., Gracias, N., Neumann, L., and Vågstøl, H. 2017. Real-time Fish Detection in Trawl Nets. *In* Proc. of the MTS/IEEE OCEANS 2017 Conference, Aberdeen, UK, pp. 1–5.
- Prados, R., Garcia, R., and Neumann, L. 2014. Image Blending Techniques and Their Application in Underwater Mosaicing, Springer, Heidelberg. ISBN: 978-3-319-05557-2.
- Qin, H., Li, X., Liang, J., Peng, Y., and Zhang, C. 2016. DeepFish: accurate underwater live fish recognition with a deep architecture. *Neurocomputing*, 187: 49–58.
- Qin, H., Peng, Y., and Li, X. 2014. Foreground extraction of underwater videos via sparse and low-rank matrix decomposition. *In* ICPR Workshop on Computer Vision for Analysis of Underwater Imagery, Stockholm, 2014, pp. 65–72. DOI: 10.1109/CVAUI.2014.16.
- Ren, S., He, K., Girshick, R., and Sun, J. F. 2015. Faster R-CNN: towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39: 1137–1149.
- Rosen, S., and Holst, J. C. 2013. DeepVision in-trawl imaging: sampling the water column in four dimensions. *Fisheries Research*, 148: 64–73.
- Sanchez-Torres, G., Ceballos-Arroyo, A., and Robles-Serrano, S. 2018. Automatic measurement of fish weight and size by processing underwater hatchery images. *Engineering Letters*, 26: 461–472.
- Schwarz, M. W., Cowan, W. B., and Beatty, J. C. 1987. An experimental comparison of RGB, YIQ, LAB, HSV, and opponent color models. *ACM Transactions on Graphics*, 6: 123–158.
- Spampinato, C., Giordano, D., Di Salvo, R., Chen-Burger, Y.-H. J., Fisher, R. B., and Nadarajan, G. 2010. Automatic fish classification for underwater species behavior understanding. *In* Proceedings of the First ACM International Workshop on Analysis and Retrieval of Tracked Events and Motion in Imagery Streams, ACM, Firenze, Italy, pp. 45–50.

Handling editor: Cigdem Beyan

Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Original Article

Detecting and counting harvested fish and identifying fish types in electronic monitoring system videos using deep convolutional neural networks

Chi-Hsuan Tseng¹ and Yan-Fu Kuo ^{1*}

¹Department of Biomechatronics Engineering, National Taiwan University, Taipei, Taiwan

*Corresponding author: tel: + 886 2 3366 5329; fax: + 886 2 2362 7620; e-mail: ykuo@ntu.edu.tw.

Tseng, C.-H. and Kuo, Y.-F. Detecting and counting harvested fish and identifying fish types in electronic monitoring system videos using deep convolutional neural networks. – ICES Journal of Marine Science, 77: 1367–1378.

Received 15 November 2019; revised 24 March 2020; accepted 3 April 2020; advance access publication 27 May 2020.

The statistics of harvested fish are key indicators for marine resource management and sustainability. Electronic monitoring systems (EMSs) are used to record the fishing practices of vessels in recent years. The statistics of the harvested fish in the EMS videos are manually read and recorded later by operators in data centres. However, this manual recording is time consuming and labour intensive. This study proposed an automatic approach for prescreening harvested fish in the EMS videos using convolutional neural networks (CNNs). In this study, harvested fish in the frames of the EMS videos were detected and segmented from the background at the pixel level using mask regional-based CNN (mask R-CNN). The number of the fish was determined using time thresholding and distance thresholding methods. Subsequently, the types and body lengths of the fish were determined using the confidence scores and the masks predicted by the mask R-CNN model, respectively. The trained mask R-CNN model attained a recall of 97.58% and a mean average precision of 93.51% in terms of fish detection. The proposed method for fish counting attained a recall of 93.84% and a precision of 77.31%. An overall accuracy of 98.06% was obtained for fish type identification.

Keywords: convolutional neural networks, fish body length, fish type identification, instance segmentation, resource management

Introduction

Fish is a primary source of food. In 2016, >90 million tonnes of fish were harvested globally and the economic value of the fishery industry reached US\$130 billion (FAO, 2018). Fishery management plays a crucial role in sustaining marine resources. Regional fisheries management organizations require fishing vessels to report the statistics of harvested fish (FAO, 2017). Conventionally, the reported statistics are recorded manually by observers or fishermen on vessels. Manual recording is time consuming and increases

the workload of fishermen. Thus, some countries have implemented electronic monitoring systems (EMSs; Ames *et al.*, 2007; Kindt-Larsen *et al.*, 2011; Needle *et al.*, 2015; Bartholomew *et al.*, 2018) on vessels in recent years to record the fishing process. Then, the information pertaining to the harvested fish [e.g. fish count, type, and body length (BL)] in the EMS images or videos are manually screened and analysed in offshore data centres. However, manual screening of the EMS images and videos is time consuming and labour intensive (Needle *et al.*, 2015; Van Helmond *et al.*,

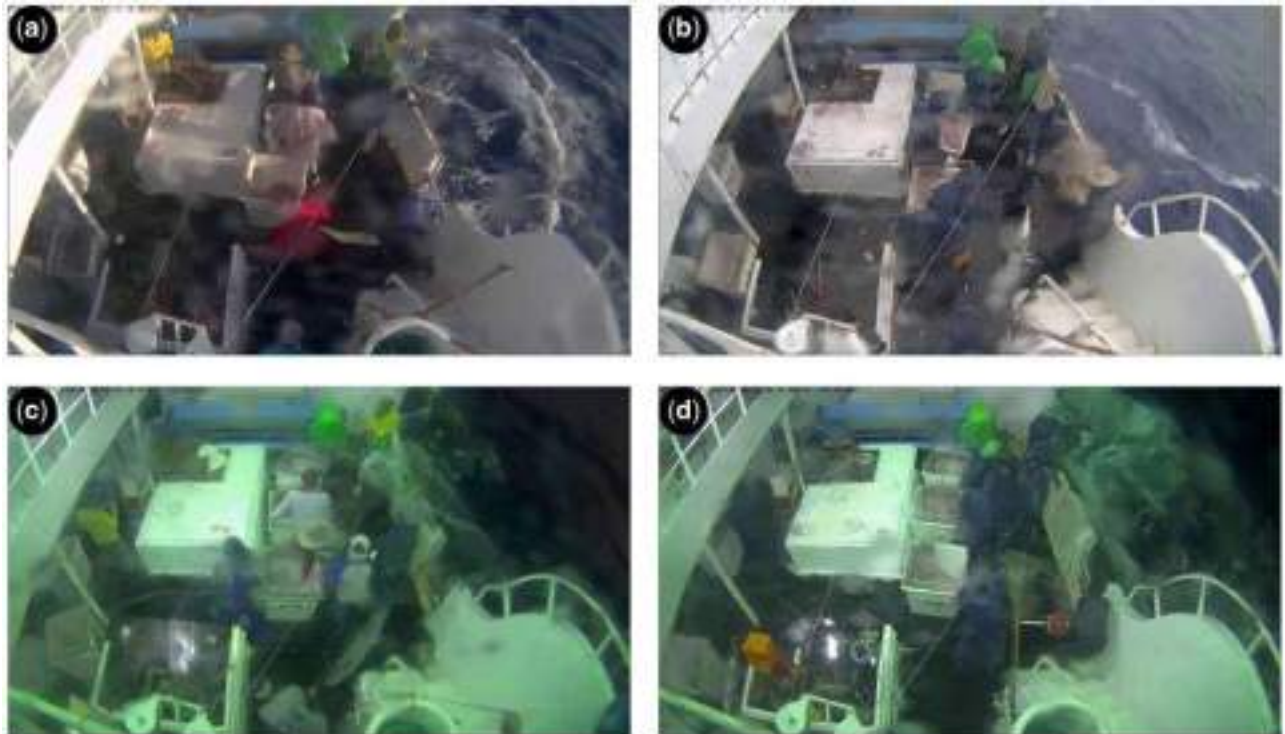


Figure 1. Video images acquired using an EMS on a longliner under: (a) sunny day, (b) rainy day, (c) dark night, and (d) rainy night.

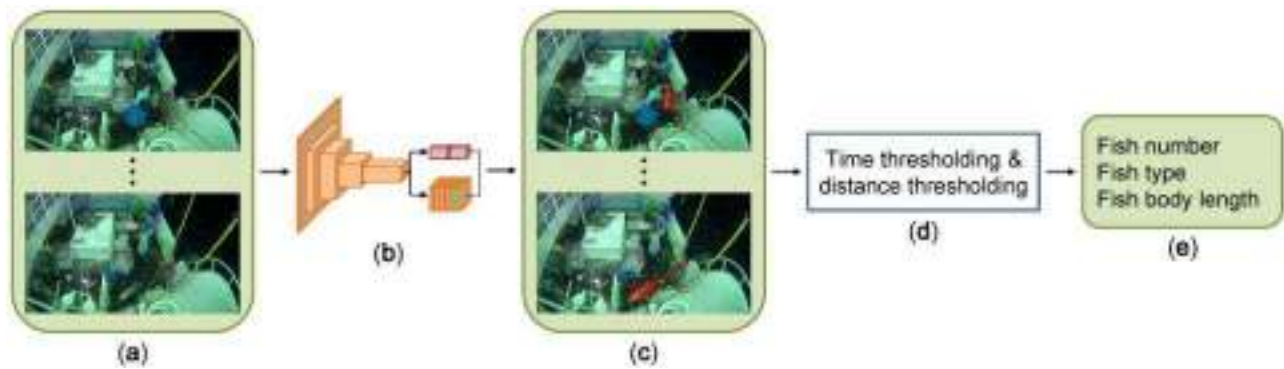


Figure 2. Flowchart of collection of the statistics of harvested fish from an EMS video: (a) EMS video frames; (b) mask R-CNN model; (c) fish candidates in the frames; (d) time and distance thresholding; and (e) fish counting, type identification, and BL estimation.

2017). Moreover, misreporting could occur due to human fatigue. To reduce the workload of interpreters, an automated approach is required for prescreening the EMS images and videos and for enabling the recording of the harvested fish statistics.

The prescreening tasks of EMS images and videos are as follows: (i) detecting harvested fish in the videos, (ii) counting the number of fish, (iii) identifying the types of the fish, and (iv) estimating the lengths of the fish. Studies have been conducted to automatically fulfil the aforementioned tasks using conventional image processing and machine learning techniques. White *et al.* (2006) detected fish and determined the lengths and species of the fish using a conveyor belt-based machine vision system. Larsen *et al.* (2009) identified three types of fish in images acquired under controlled illumination using manually annotated

shape and texture features and linear discriminant analysis. Shafry *et al.* (2012) evaluated the BLs of fish in images acquired with a white background using image processing techniques. Morais *et al.* (2005) tracked and counted fish in underwater images acquired in a laboratory using machine vision techniques. Spampinato *et al.* (2008) detected, tracked, and counted fish in coral reef videos collected with a high-contrast background using moving average and feature-matching algorithms. Toh *et al.* (2009) detected and counted feeder fish in static containers using image processing techniques. Although all the aforementioned approaches achieved excellent performance, they required the images of fish to be acquired under specific conditions. However, EMS images or videos acquired from vessels are complex (Figure 1). Fishing is conducted during the day, the night, sunny

Table 1. Statistics of the weather conditions of the fish harvesting videos.

	Sunny day	Dark night	Rainy day	Rainy night	Total
Training and validation	80	240	45	135	500
Test	18	83	25	74	200
Total	98	323	70	209	700

Table 2. Number of the training and validation images and labelled objects.

	Training		Validation	
	Image	Object	Image	Object
Tuna	1 600	1 769	400	440
Marlin	400	428	100	104
Shark	400	409	100	107
Others	400	478	100	104
Buoy	400	441	100	106
Total	3 200	3 525	800	861

days, and rainy days, and the illumination conditions are uncontrollable. The acquired images and videos can be blurred by the presence of water drops on the lens of the camera. Moreover, the decks of fishing vessels are usually filled with miscellaneous items. Thus, the image processing and machine learning-based approaches may be suboptimal for detecting fish in EMS images or videos.

Recently, convolutional neural networks (CNNs) have been applied for solving complex machine vision problems (LeCun *et al.*, 1998; Krizhevsky *et al.*, 2012). CNNs are derivatives of multilayer perceptrons that utilize spatial convolutional operations in networks. Once trained, CNNs can extract features from input images with almost no preprocessing requirements and can achieve remarkable performance in machine vision tasks (e.g. classification and localization). Studies have employed CNN approaches for solving fish detection and recognition problems. French *et al.* (2015) detected and counted discarded harvested fish in fishing trawler videos using CNNs and the nearest neighbour algorithm. Li *et al.* (2016) detected fish and identified fish species in underwater videos using faster regional-based CNN (R-CNN; Ren *et al.*, 2015). Qin *et al.* (2016) recognized fish in underwater videos through CNNs, matrix decomposition, and support vector machine. Zhuang *et al.* (2017) completed three fish recognition tasks in coral reef videos using CNNs of single shot multibox detector architecture (Liu *et al.*, 2016). Lu *et al.* (2019) identified eight common species and types of harvested tuna and billfish using CNNs of VGG architecture (Simonyan and Zisserman, 2014). Sung *et al.* (2017) located fish in unconstrained underwater videos using CNNs of you-only-look-once architecture (Redmon *et al.*, 2016). Jäger *et al.* (2017) tracked the movements of multiple fish in streams in underwater videos using a two-stage graph-based approach and CNN models. Zheng *et al.* (2018) detected fish and recognized the species of the fish in images acquired from vessels using the local region-based modelling approach and deep CNNs. Tseng *et al.* (2020) measured fish BL using CNN in images acquired on vessels. Another work detected fish in images and estimated the lengths of the fish using three R-CNNs (Monkman *et al.*, 2019).

The first essential step in identifying the types of fish and estimating the lengths of the fish involves localization and segmentation of fish in images. Mask R-CNN (He *et al.*, 2017), which combines the localization function of a faster R-CNN and the pixel-wise segmentation function from a fully convolutional network (Long *et al.*, 2015), satisfies this requirement. Several examples of this technique are being used in marine sciences. Ditria *et al.* (2019) employed the mask R-CNN framework for detecting luderick (*Girella tricuspidata*) in underwater images for fish abundance quantification. Francisco *et al.* (2019) detected animals in three-dimensional underwater images and estimated the BLs of the animals at the pixel level by using mask R-CNN. One of the studies that applied mask R-CNN to fishery was conducted by French *et al.* (2019). They counted and identified the species of discarded harvested fish in closed-circuit television videos acquired from fishing trawlers using mask R-CNN. Garcia *et al.* (2019) utilized mask R-CNN to localize and segment fish in images captured during commercial trawling for fish size measurement. Another study estimated the lengths of boxed fish at port using mask R-CNN and unsupervised learning approach (Álvarez-Ellacuría *et al.*, 2019).

This study proposed a novel procedure to prescreen fishing videos and to collect statistics of the harvested fish in the videos. The statistics included fish count, type, and BL in pixels. The proposed approach used an existing EMS to acquire fishing videos. Operators only have to verify the fish detected by the proposed procedure, instead of watching the entire videos. Thus, hundreds of labour hours could be saved. In the proposed procedure, EMS videos acquired from longliners were converted to images using a rate of one frame per second (fps). A mask R-CNN model was developed and used to automatically detect and segment the fish from the background at the pixel level (Figure 2b). Subsequently, the fish were counted using time thresholding and distance thresholding to remove false-positive detections and avoid double counting the detected fish in sequential frames (Figure 2d). The types (i.e. tuna, marlin, shark, or others) and BLs of the fish were also determined using the information obtained from the mask R-CNN model.

Material and methods

Image collection and training data preprocessing

Seven hundred videos pertaining to fish harvesting were provided by the Fisheries Agency, Council of Agriculture, Taiwan. The videos were acquired on the deck of a longliner using SeaTube (Satlink, Madrid, Spain) in 2018. Each video had a length of 10 min. The frame rate and resolution of the videos were 30 fps and 1280 × 720 pixels, respectively. The videos were acquired under uncontrolled weather conditions (e.g. sunny days, rainy days, and dark nights; Figure 1 and Table 1). A total of 500 videos were used for training and validating the mask R-CNN model for fish detection and segmentation. The remaining 200 videos were used

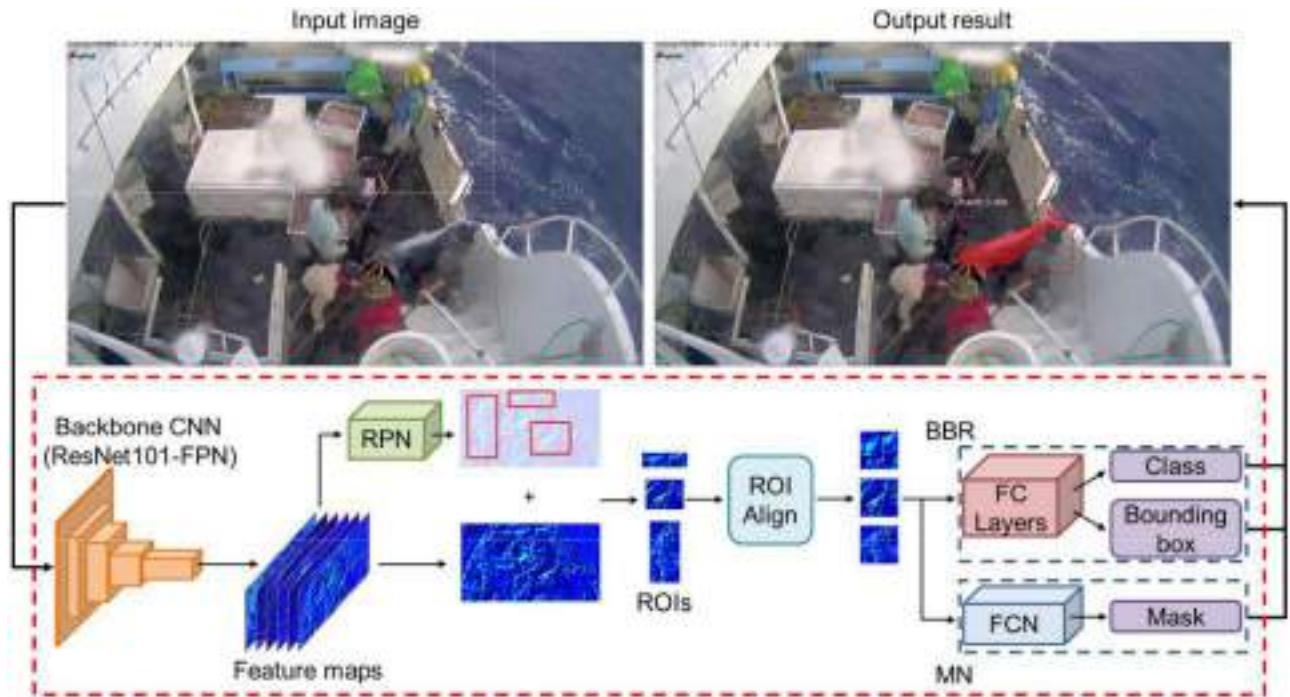


Figure 3. Architecture of the mask R-CNN. The outputs of the mask R-CNN include the class (tuna, marlin, shark, other fish, or buoy), bounding box, and mask (red filled area of the fish body) of the object.

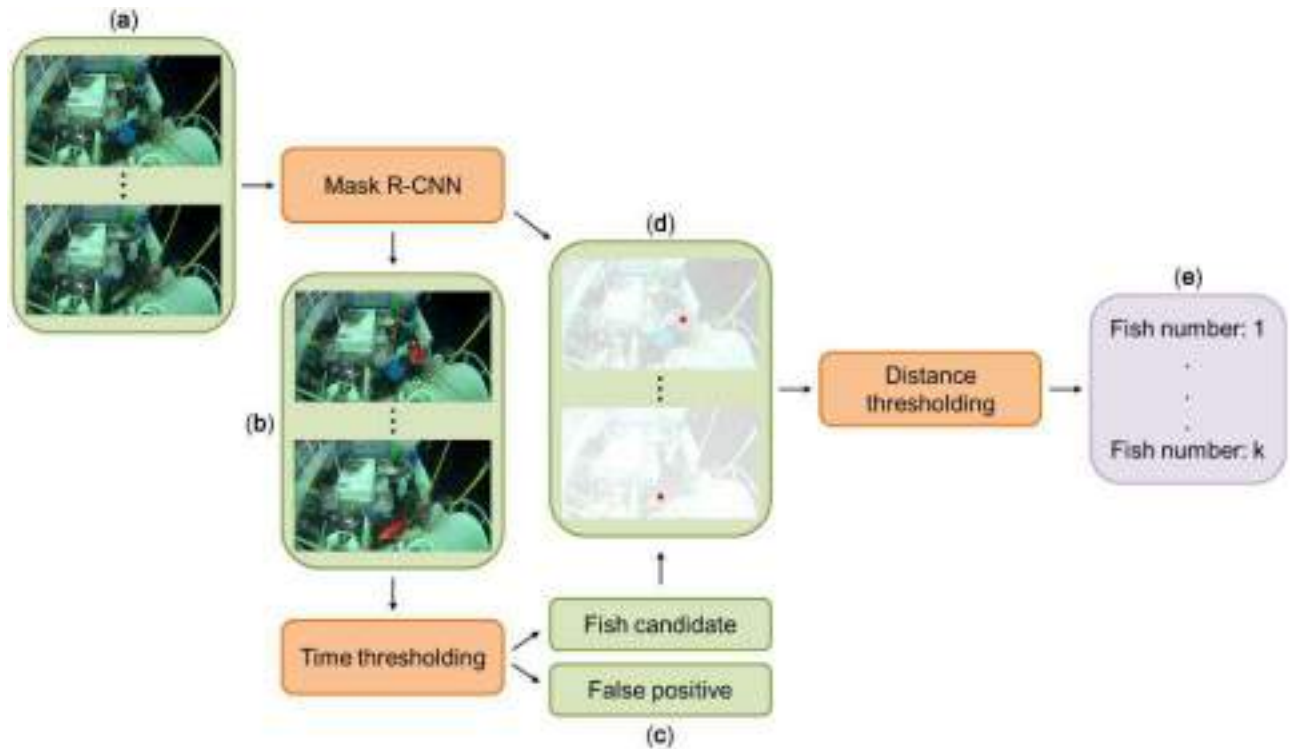


Figure 4. Fish detection and counting in the videos: (a) consecutive input frames, (b) frames of fish candidates, (c) false-positive removal, (d) geometric centre of the true-positive fish candidates, and (e) counting the number of fish in the video.

for assessing the proposed fish counting method (e.g. mask R-CNN, time thresholding, and distance thresholding). These videos were referred to as test videos.

A total of 4000 images were generated from the 500 training videos. Each image contained at least one harvested fish or buoy (Table 2). Approximately 80 and 20% of the images were used for

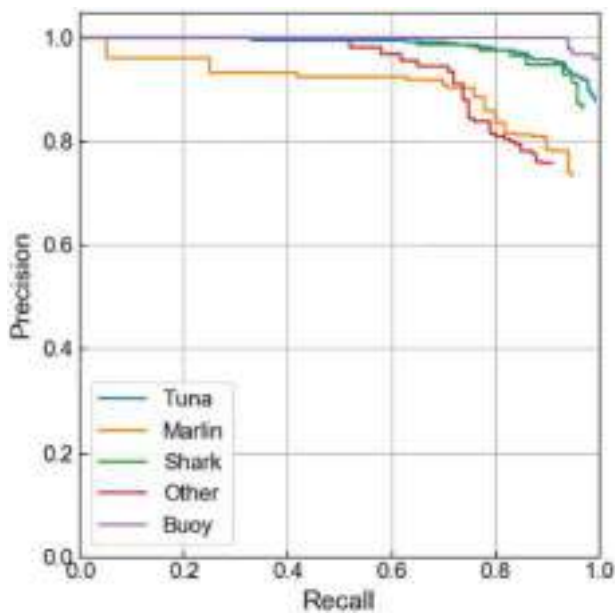


Figure 5. Precision–recall curves of the four types of fish (tuna, marlin, shark, and other) and buoy.

training and validating the mask R-CNN model, respectively. These images were referred to as training and validation images. The images were resized to 1024 × 576 pixels to reduce the processing time. The four classes of fish (i.e. tuna, marlin, shark, and other fish) and buoys in the training and test images were labelled manually using the LabelMe toolkit (Russell et al., 2008).

Mask R-CNN and training methodology

Mask R-CNN (Figure 3) was used to detect and locate objects (e.g. fish or buoys) in the images. The mask R-CNN model comprised a feature extractor (FE), region proposal network (RPN; Ren et al., 2015), bounding box recognition (BBR), and mask network (MN). The FE generated feature maps from an input image. The RPN then suggested the regions of interest (ROI; i.e. regions that contain fish or buoys) in the feature maps. Next, the ROI and corresponding feature maps were input into the BBR to determine the classes (e.g. tuna, marlin, shark, other fish, or buoys) and the precise bounding boxes of the objects. The ROI and corresponding feature maps were also input into the MN to generate binary masks for the objects. The binary masks were pixel-wise foreground layers of the objects located in the ROI (Figure 3).

In this study, a combination of ResNet101 (He et al., 2016) and a feature pyramid network (Lin et al., 2017) was used as the FE (see Supplementary Material for more details of the FE). The settings of the RPN are described as follows. The RPN proposed ROIs of four scales (32 × 32, 64 × 64, 128 × 128, and 256 × 256 pixels) and three aspect ratios (1:2, 1:1, and 2:1). The ROIs were positive or negative. An ROI was defined as positive if the ROI and any ground-truth object (i.e. fish or buoy) had an intersection over union (IoU) value >0.7. An ROI was defined as negative if the ROI and all the ground-truth objects had IoU values <0.3. Nonmaximum suppression (NMS; Neubeck and Van Gool, 2006) was applied to reduce the ROI proposals of the same

Table 3. Receiver-operating characteristic analysis of the developed mask R-CNN model.

Type	Precision (%)	Recall (%)	F1-score (%)	AP (%)
Tuna	88.04	99.30	93.33	98.10
Marlin	73.64	95.00	82.96	86.98
Shark	86.60	97.00	91.50	95.89
Others	75.83	91.00	82.72	86.79
Buoy	96.15	100.00	98.03	99.82
Overall	86.91	97.58	91.93	93.51

object. The threshold of NMS was set to 0.7 (see Supplementary Material for more details of the BBR and MN).

The training of the mask R-CNN model was conducted as follows. The model was first initialized with the parameters pre-trained using the COCO dataset (Lin et al., 2014). The BBR and MN were trained for 1000 epochs using a minibatch of eight training images. During the training, the parameters of the FE and RPN were fixed. Subsequently, FE, RPN, BBR, and MN were fine-tuned for another 1000 epochs using a minibatch of four training images. At each epoch, image augmentation was applied to the images for enhancing the model performance. The augmentation operations included rotation (randomly rotated between −15° and 15°), horizontal and vertical shifting (randomly shifted one-tenth of the width or height), brightness variation (randomly multiplied between 0.8 and 1.2), blurring (using a Gaussian kernel with a sigma of 3), and scaling (randomly scaled between 0.8 and 1.2). Zero, one, or two operations were randomly applied to each training image. The RPN was set to provide 200 ROIs. The ratio of positive to negative ROIs obtained from the training images was set to 1:3. Stochastic gradient descent (Bottou, 2010) was used as the optimizer. The initial learning rate, momentum, and weight decay were set to 0.001, 0.9, and 0.0001, respectively. The model was trained using an open-source Python environment (Van Rossum and Drake, 1995) and Tensorflow (Abadi et al., 2016) and Keras (Chollet, 2015). Two graphic processing units (GPU; GeForce GTX 1080Ti, Nvidia, Santa Clara, CA, USA) were used to expedite the training process.

Fish counting in the videos

The fish were counted using the developed mask R-CNN model, time thresholding, and distance thresholding (Figure 4). In the procedure, a video (10 min) was first converted to images using a rate of 1 fps. Then, the 600 frames of the video were fed into the developed mask R-CNN model sequentially for detecting fish (Figure 4a). If an object in a frame was detected as fish, then the object was segmented at the pixel level (Figure 4b) and was labelled as a fish candidate of the frame. The geometric centre of the fish candidate was located, and the confidence score and class of the fish candidate from the mask R-CNN model were also recorded. After the completion of fish detection on all the 600 frames, time thresholding was applied to remove false-positive fish candidates (Figure 4c). The fish candidate in a frame was considered as a false positive if no fish was detected in the following five frames (5 s). Subsequently, distance thresholding was applied to determine whether the fish candidates presented in consecutive frames were the same fish. In the procedure, the distance between the geometric centres of the fish candidates in two consecutive frames (Figure 4d) was calculated. If the distance was less than a threshold, then the fish candidates were considered the

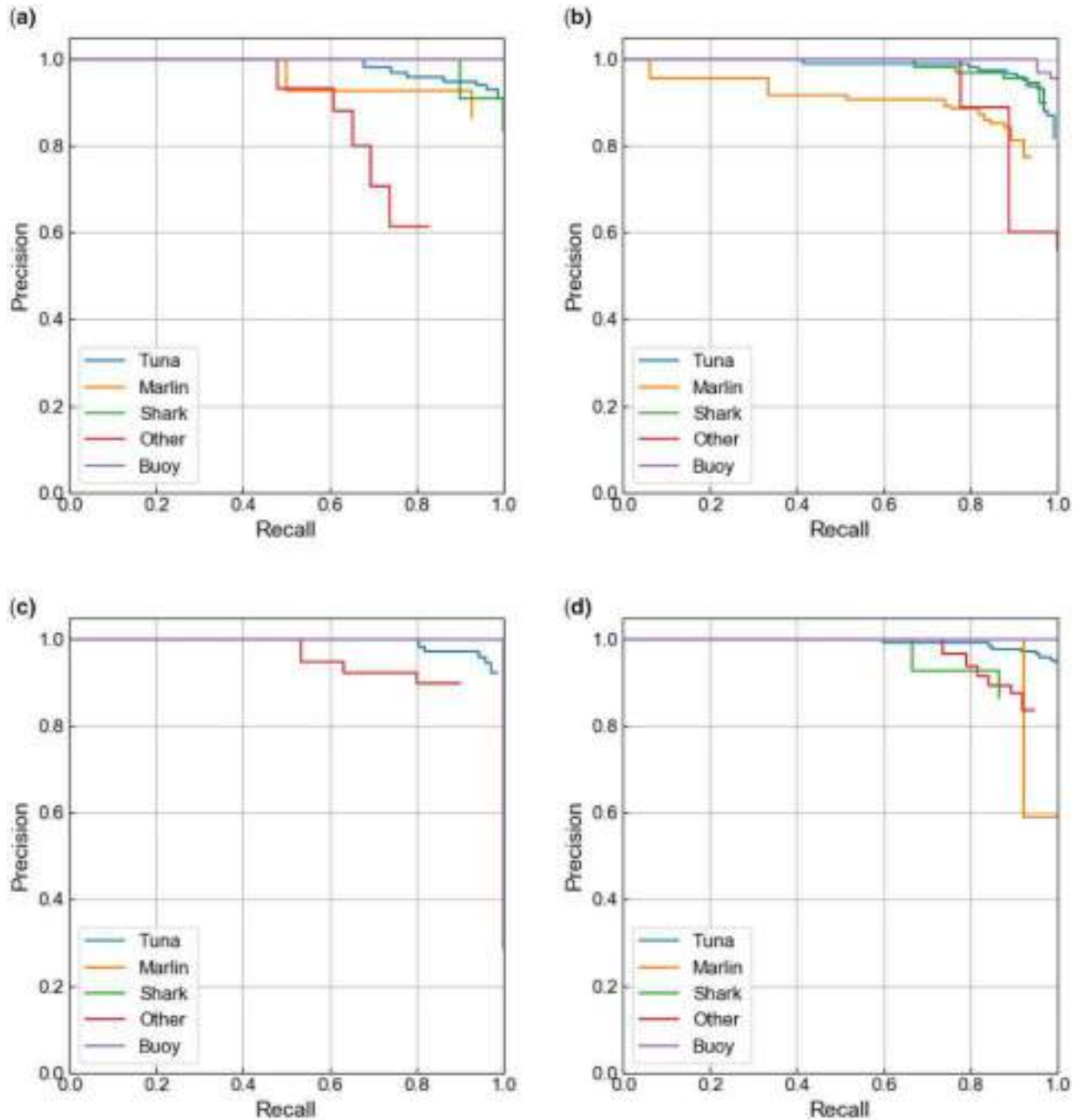


Figure 6. Precision–recall curves of the four weather conditions: (a) sunny day, (b) dark night, (c) rainy day, and (d) rainy night.

same (Figure 4e). Otherwise, the fish candidates were considered different. In this study, the threshold was set to 125 pixels.

Fish type identification and BL measurement

After the number of fish was counted, the type of each counted fish was determined using five instances of the fish candidates with the top five confidence scores and majority rule. The BL of a fish was also determined as the mean BL of the five instances. The BL of an instance was measured to be the distance in pixels between the farthest ends of the fish

mask. To prevent the occlusion cases from being included, the BL of an instance was not used if the length of the instance was smaller than 85% of the mean BL of the five instances.

Results and discussion

Training loss of the mask R-CNN model

The training, training mask, training bounding box, training classification, validation, validation mask, validation bounding box, and validation classification losses of the mask R-CNN model

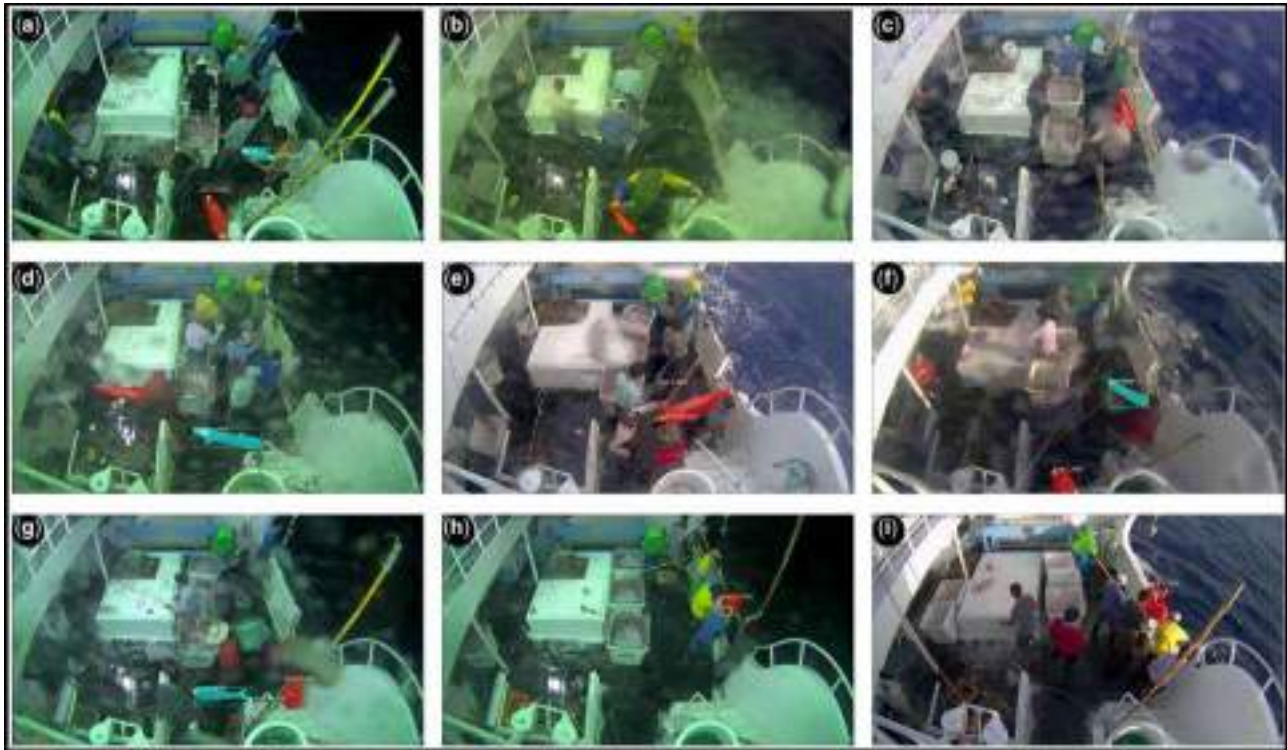


Figure 7. Successful detection of fish and buoys in the validation images. The first and second detected objects are coloured in red and cyan, respectively.

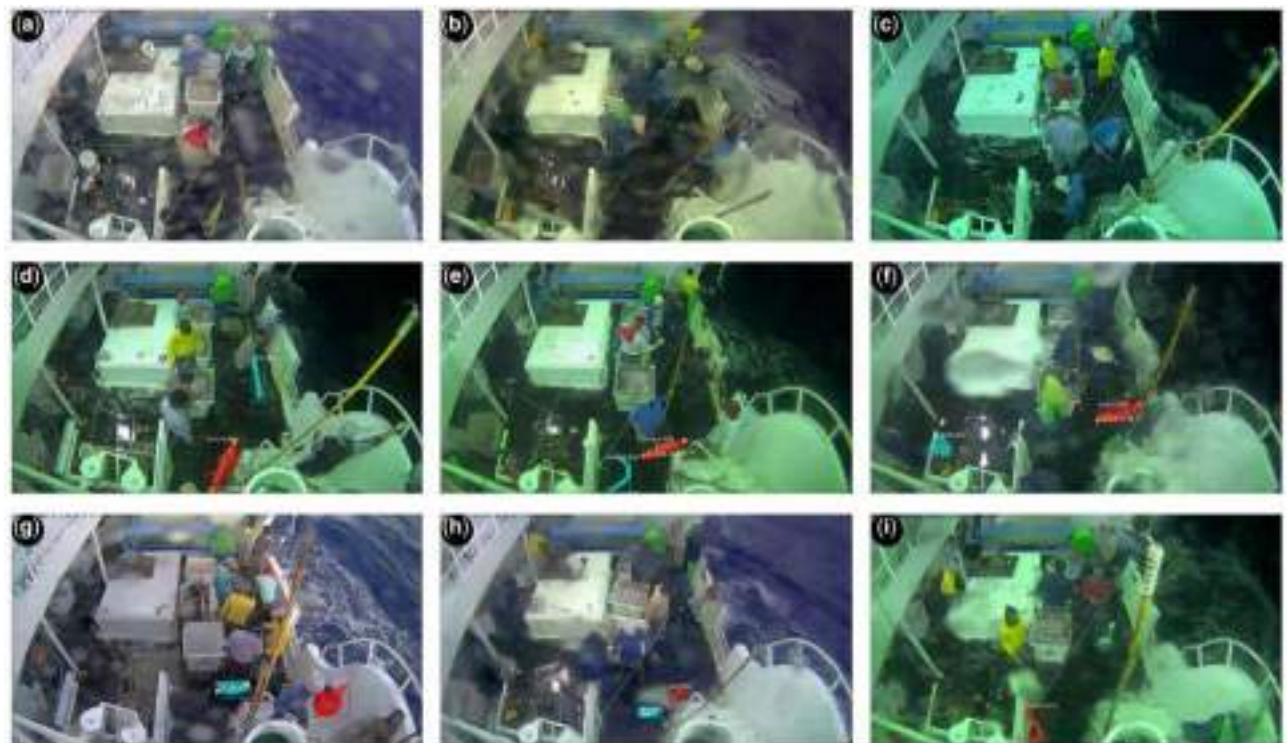


Figure 8. (a–c) False-negative detections; (d–h) false-positive detections; and (i) TP detection. (d) A bamboo hook misidentified as a fish, (e) the rope misidentified as fish, (f) a miscellaneous item misidentified as a buoy, (g) a water drop on the lens misidentified as a tuna, and (h) one fish occluded by a water drop on the lens misidentified as two fish.

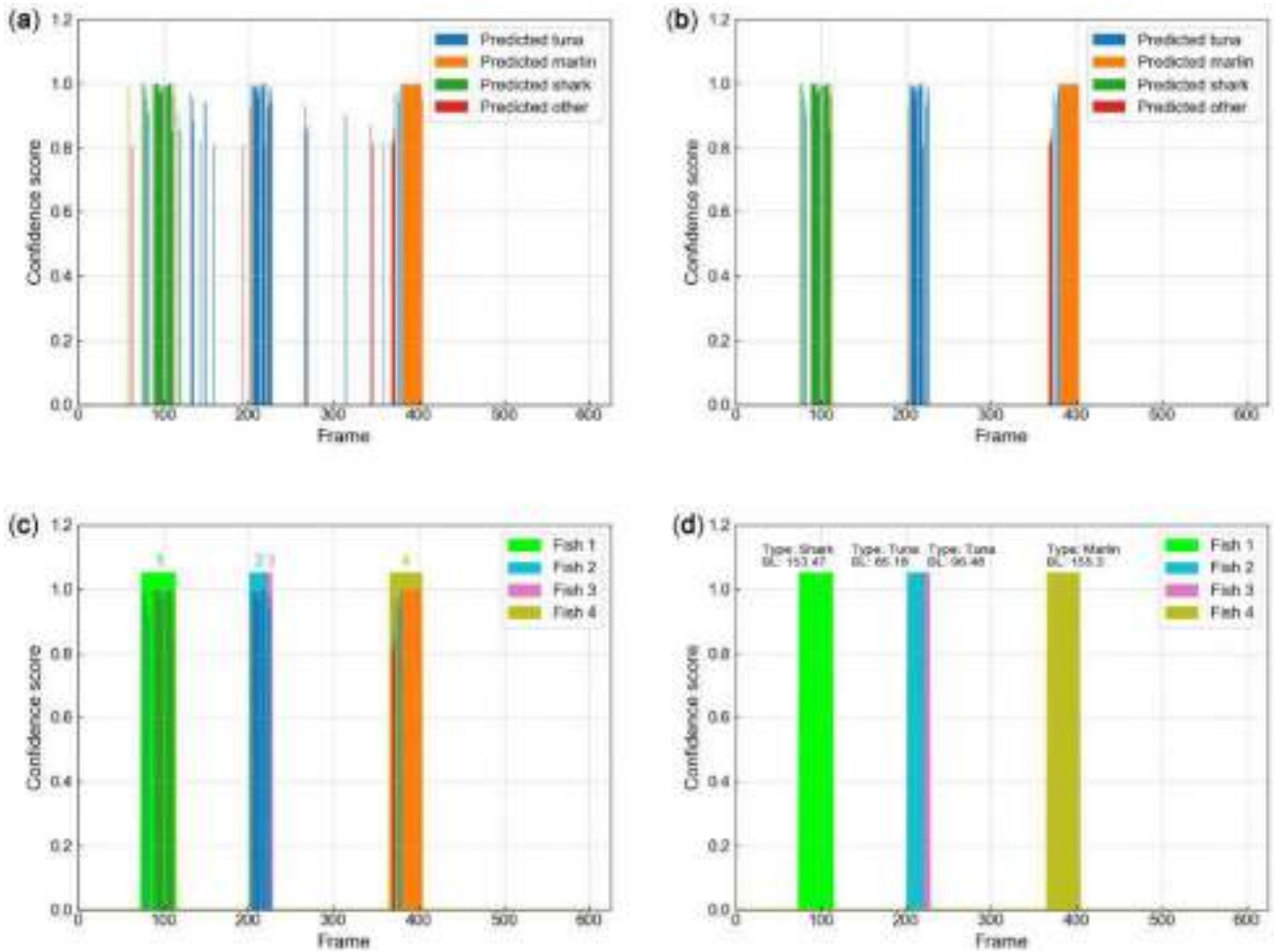


Figure 9. Results of fish counting: (a) fish detection using the mask R-CNN model, (b) false-positive removal using time thresholding, (c) fish counted using distance thresholding, and (d) fish type identification and BL estimation.

Table 4. Analysis of the proposed fish counting method.

	Ground truth	TP	FP	FN	Precision (%)	Recall (%)	Error (%)
Fish	276	259	76	17	77.31	93.84	21.37

The unit for count, TP, FP, and FN is number of fish.
 FP, false-positive count; FN, false-negative count.

Table 5. Analysis of fish frame detection.

	True positive	Precision (%)	Recall (%)	Average frame error			
				Start (F)	End (F)	Total (F)	Total (%)
Fish	259	96.38	99.33	1.07	1.55	2.62	5.95

F, frame.

were examined. After 1000 epochs of the BBR and MN training, the training and validation losses converged to 0.4 and 0.65, respectively. After 1000 epochs of all training layers, the training and validation losses converged to 0.25 and 0.35, respectively. These observations indicate that the model learned the features of the fish and buoys in the videos.

Performance of fish and buoy detection

The performance of the developed mask R-CNN model was assessed using the receiver-operating characteristic analysis (Fawcett, 2006) and the 800 validation images (Table 3). In the analysis, the confidence score threshold of the developed mask R-CNN model was set to 0.8. The model attained an overall precision of 86.91% and an overall recall of 97.58% (Table 1). The high recall of 97.58% indicated that the trained model could avoid false identification of a fish as not a fish. This observation indicated that the developed mask R-CNN model can be useful for prescreening EMS videos. Certain items on the deck may be misidentified as fish because of the mediocre precision of 86.91%. Nevertheless, the false-positive detections could later be verified and removed by operators. Figure 5 illustrates the precision–recall curves (Manning et al., 1999) used for detecting the four types of fish and buoys by using the validation images. The developed mask R-CNN model attained a mean average precision (AP; Everingham and Winn, 2011) of 93.51%.

Performance of fish and buoy detection under various weather conditions

The performance of the developed mask R-CNN model under various weather conditions was also assessed (Figure 6) using the

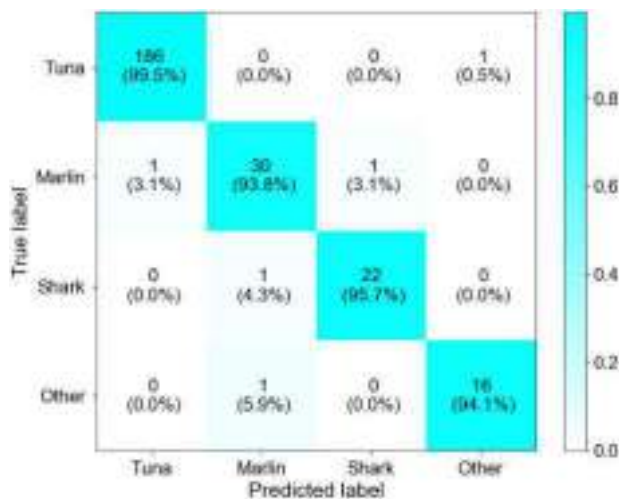


Figure 10. Confusion matrix for fish type identification based on TPs.

800 validation images. The validation set contained 140, 320, 110, and 230 of sunny day, dark night, rainy day, and rainy night images, respectively. The trained mask R-CNN model achieved recalls of 96.64, 98.03, 95.57, and 98.21% on sunny day, dark night, rainy day, and rainy night images, respectively. The results indicated that the trained mask R-CNN model could be utilized under various weather conditions. The recalls of dark night and rainy night were slightly better than those of sunny day and rainy day because there were more training images of dark night and rainy night (Table 1). We believe that increasing the number of training images can further improve the performance of the model.

Successful case study of fish and buoy detection

Figure 7 illustrates the successful detection of fish and buoys in the validation images. The images had a high complexity level. Miscellaneous items and fishermen filled the deck. Moreover, the weather conditions of the images varied considerably, including dark nights (Figure 7a, b, d, g, and h), sunny days (Figure 7c, e, and i), and high contrast (Figure 7f). Furthermore, sea water drops on the lens caused a blur in some images (Figure 7b, c, d, f, and g). Despite these challenges, the trained model correctly detected and segmented tuna (Figure 7a, f, and 7g), marlin (Figure 7d and e), common dolphinfish (*Coryphaena hippurus*; Figure 7b, c, and f), and buoys (Figure 7g–i). The model also detected multiple objects presented in an image (Figure 7a, d, f, and g).

Failure case study of fish and buoy detection

Images with unsuccessful detection of fish and buoys were examined. False negatives of fish detection occurred due to an occlusion by a fisherman (Figure 8a), image blur caused by water drops on the lens (Figure 8b), or pose change by a fisherman (Figure 8c). False positives of fish and buoy detections are illustrated in rows 2 and 3 of Figure 8 (except for Figure 8i). Certain items on the deck were misidentified as fish (e.g. bamboo hook, the cyan object presented in Figure 8d, and tube, the cyan object presented in Figure 8e) or buoy (e.g. a miscellaneous item, the cyan object displayed in Figure 8f). Certain water drops on the

Table 6. Analysis of the proposed fish type identification for the four types of fish based on ground truth.

Class	Count	TP	FP	FN	Precision (%)	Recall (%)	F1-score (%)	Error (%)
Tuna	195	186	40	9	82.30	95.38	88.35	15.89
Marlin	35	30	8	5	78.94	85.71	82.18	8.57
Shark	26	22	9	4	70.96	84.61	77.18	19.23
Others	20	16	24	4	40.00	80.00	53.33	100.00
Overall	276	254	81	22	75.82	92.02	83.13	21.37

lens were also misidentified as fish (the cyan object presented in Figure 8g) or the same fish was detected as two fish (Figure 8h). Figure 8e displays a false-positive detection case of fish (in cyan). The shape of the false-positive object highly resembled a true-positive (TP) object in Figure 8i. The differences between the TP and false-positive objects can be subtle. Nevertheless, these false-positive objects were usually isolated in certain frames and could be filtered out using the subsequent time thresholding method.

Fish counting performance

The proposed fish counting approach is illustrated using a fish video (Figure 9). In the procedure, fish candidates in the 600 frames of the video were detected using the trained mask R-CNN model (Figure 9a). Then, false-positive fish candidates were removed using time thresholding (Figure 9b). Subsequently, the fish candidates present in the consecutive frames were determined to be the same fish by using distance thresholding (Figure 9c). Finally, the type and BLs of each fish (Figure 9d) were then determined based on the majority class and the mean BL, respectively, of the fish candidates that correspond to the top five confidence scores of the fish. The mean processing speeds of fish detection and counting were 3 and 1.6 fps, respectively, and the processing was conducted using the GPU.

The performance of the proposed fish counting approach was evaluated using the 200 test videos. The operators verified that the videos contained a total of 276 ground-truth fish. The proposed method attained a precision of 77.31% and a recall of 93.84% (Table 4). The high recall and intermediate precision indicated that the proposed approach can be useful for prescreening EMS videos. The false-positive detections could later be verified and removed by operators.

The frame detections of the 259 TP fish were evaluated (Table 4). The 259 TP fish contained a total of 20 666 frames. For each TP, the corresponding fish frames ranged between 5 and 552 frames. The proposed approach obtained a precision of 96.38% and a recall of 99.33% in frame detection (Table 5). The start, end, and total frame errors of the 259 TP fish were also evaluated. The start frame error of a fish was defined as the difference between the start frame predicted by the proposed approach and the ground-truth start frame. Similarly, the end frame error of a fish was defined as the difference between the end frame predicted by the proposed approach and the ground-truth end frame. The total frame error was the summation of the start and end frame errors. The mean total frame error was 2.62 frames (5.95%; Table 5).

Performance of fish type identification

The performance of the proposed method in terms of fish type identification was evaluated using the 200 test videos. According

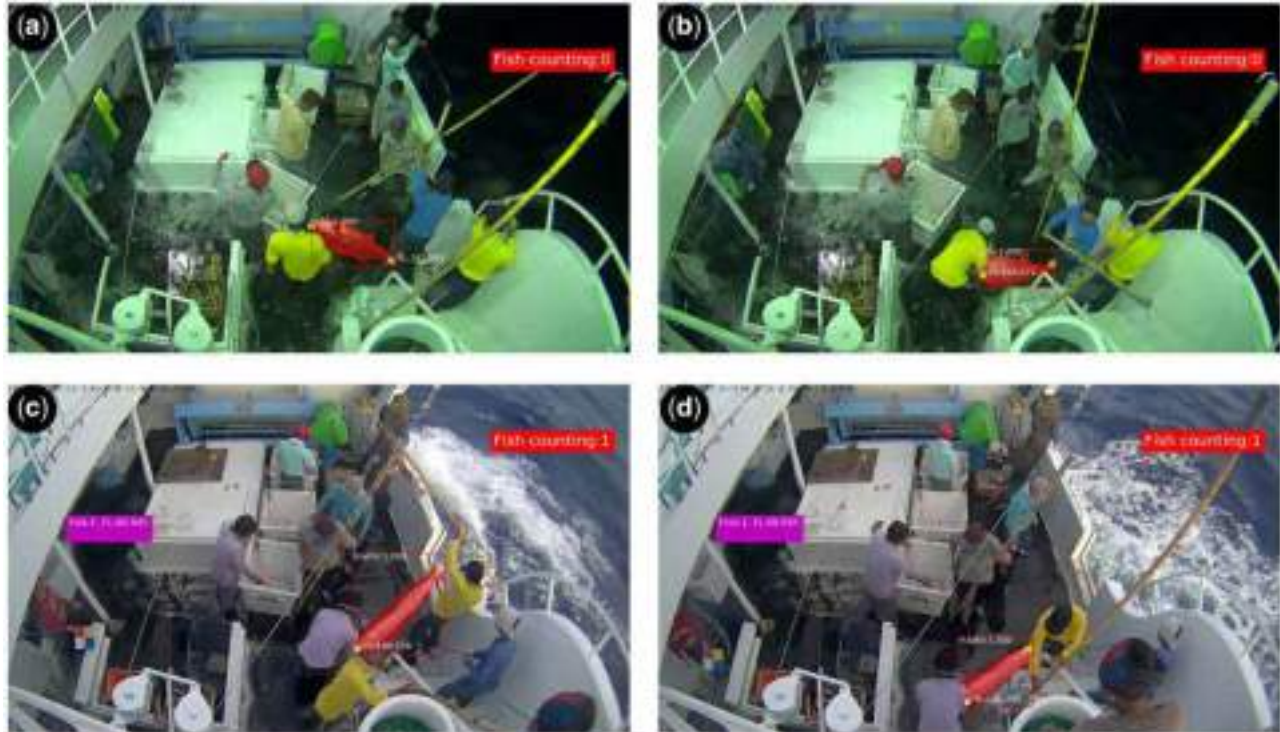


Figure 11. Effect of occlusion in fish BL estimation: (a and c) complete fish bodies and (b and d) fish bodies partially occluded by fishermen. The bodies in (a) and (b) pertain to the same fish. The bodies in (c) and (d) are of the same fish. The underestimated BLs in (b) and (d) were filtered out by the proposed 85% rule.

to the ground truths (276 in Table 4), the proposed approach attained an overall precision of 75.82%, the recall of 92.02%, and the F1-score of 83.13% (Table 6). The trained model misidentified certain miscellaneous objects as other fish, thus causing low precision and high error of the other fish class. However, this type of false-positive errors can be corrected by operators when they verify the prescreening results. By only considering the TP detections (259 TPs in Table 4), the fish type identification reached an overall accuracy of 98.06% (Figure 10).

Performance of fish BL estimation

The performance of the proposed method in terms of fish BL estimation was evaluated using the 259 TP fish. The 259 TP fish were associated with 1295 frames (five frames per fish corresponding to the top five confidence scores). Among the frames, the fish bodies in 118 frames were occluded (Figure 11b and d). The proposed 85% rule excluded 95.53% of the 118 frames of occluded fish bodies from being used in BL estimation. Thus, the performance of BL estimation was improved. The fish BL obtained in pixels could be converted to the fish lengths in physical units if an EMS camera was calibrated during installation.

Potential implications of the proposed approach

The proposed approach could potentially be used in data centres to assist the reporting of fishing statistics. In some practices, the videos of fishing trips recorded using EMSs are transferred to data centres after fishing vessels return to ports. The reporting of fish counts, types, and BLs in the EMS videos is then manually performed. The proposed approach could be used to prescreen

harvested fish in the EMS videos. Operators then can verify the results annotated by the proposed algorithm and generate the statistics of the harvested fish semi-automatically. Hundreds of hours of manually analysing EMS videos could be saved. The proposed approach is compatible with existing EMSs. No modification in EMSs is required.

Conclusion and future work

This study proposed an automated method for counting harvested fish, estimating the types of the fish, and measuring the BLs of the fish in EMS videos. The method applied mask R-CNN to detect fish in the video frames, time thresholding to remove false-positive detections and distance thresholding to avoid double counting fish. The type of fish was determined based on majority rule and the types of the fish in the frames with the top five confidence scores predicted by the mask R-CNN model. The BL of fish was determined as the mean BL of the fish in the frames corresponding to the top five confidence scores. The trained mask R-CNN model attained a recall of 97.58% and a mean AP of 93.51% in object detection. The proposed fish counting method obtained a recall of 93.84%. The fish type identification provided an overall accuracy of 98.06% of TP fish. The high recall and accurate frame detection of the proposed method indicate that the approach can be used to automatically prescreen EMS videos; thus, the time and effort required for reporting fishing practice statistics can be reduced.

The achievement of this research can be a foundation for fully automatic fish counting. The remaining challenges include more accurate fish counting and species identification. In this study, time thresholding and distance thresholding were used for fish counting because they had high computational efficiencies.

Nevertheless, certain state-of-art tracking approaches can be applied to identify the same fish in a series of frames if computational burden is not an issue. As for fish species identification, we believe that increasing the number of annotated training samples will further improve the performance of the trained deep CNN models. We conclude that using computer vision to acquire the statistics of harvested fish from EMS videos recorded on long-liners is feasible.

Supplementary data

[Supplementary material](#) is available at the *ICESJMS* online version of the manuscript.

Funding

This study was supported by the Fisheries Agency, Council of Agriculture, Taiwan (106AS-18.1.7-FA-F1, 107AS-14.2.7-FA-F1, 108AS-13.2.7-FA-F1, and 109AS-11.2.7-FA-F1).

References

- Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., *et al.* 2016. Tensorflow: a system for large-scale machine learning. *In* 12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16), pp. 265–283. <https://www.usenix.org/conference/osdi16/technical-sessions/presentation/abadi>.
- Ames, R. T., Leaman, B. M., and Ames, K. L. 2007. Evaluation of video technology for monitoring of multispecies longline catches. *North American Journal of Fisheries Management*, 27: 955–964.
- Álvarez-Ellacuría, A., Palmer, M., Catalán, I. A., and Lisani, J. L. 2019. Image-based, unsupervised estimation of fish size from commercial landings using deep learning. *ICES Journal of Marine Science*, 77: 1330–1339.
- Bartholomew, D. C., Mangel, J. C., Alfaro-Shigueto, J., Pingo, S., Jimenez, A., and Godley, B. J. 2018. Remote electronic monitoring as a potential alternative to on-board observers in small-scale fisheries. *Biological Conservation*, 219: 35–45.
- Bottou, L. 2010. Large-scale machine learning with stochastic gradient descent. *In* *Proceedings of COMPSTAT'2010*, pp. 177–186. Physica-Verlag HD.
- Chollet, F. 2015. Keras. <https://github.com/fchollet/keras>.
- Ditria, E. M., Lopez-Marcano, S., Sievers, M. K., Jinks, E. L., Brown, C. J., and Connolly, R. M. 2019. Automating the analysis of fish abundance using object detection: optimising animal ecology with deep learning. *bioRxiv*, 805796.
- Everingham, M., and Winn, J. 2011. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Development Kit. Pattern Analysis, Statistical Modelling and Computational Learning, Technical Report.
- FAO. 2017. Seafood Traceability for Fisheries Compliance: Country-Level Support for Catch Documentation Schemes. FAO, Rome.
- FAO. 2018. The State of World Fisheries and Aquaculture 2018 (SOFIA 2018): Meeting the sustainable development goals. Food and Agriculture Organization, ROME.
- Fawcett, T. 2006. An introduction to ROC analysis. *Pattern Recognition Letters*, 27: 861–874.
- Francisco, F. A., Nührenberg, P., and Jordan, A. L. 2019. A low-cost, open-source framework for tracking and behavioural analysis of animals in aquatic ecosystems. *bioRxiv*, 571232.
- French, G., Fisher, M. H., Mackiewicz, M., and Needle, C. 2015. Convolutional neural networks for counting fish in fisheries surveillance video. *In* *Proceedings of the Machine Vision of Animals and their Behaviour (MVAB)*, 7-1.
- French, G., Mackiewicz, M., Fisher, M., Holah, H., Kilburn, R., Campbell, N., and Needle, C. 2019. Deep neural networks for analysis of fisheries surveillance video and automated monitoring of fish discards. *ICES Journal of Marine Science*, 77: 1340–1353.
- Garcia, R., Prados, R., Quintana, J., Tempelaar, A., Gracías, N., Rosen, S., Vågstøl, H. *et al.* 2019. Automatic segmentation of fish using deep learning with application to fish size measurement. *ICES Journal of Marine Science*, 77: 1354–1366.
- He, K., Gkioxari, G., Dollár, P., and Girshick, R. 2017. Mask R-CNN. *In* *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961–2969.
- He, K., Zhang, X., Ren, S., and Sun, J. 2016. Deep residual learning for image recognition. *In* *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778.
- Jäger, J., Wolff, V., Fricke-Neudert, K., Mothes, O., and Denzler, J. 2017. Visual fish tracking: combining a two-stage graph approach with CNN-features. *In* *OCEANS 2017-Aberdeen*, pp. 1–6. IEEE.
- Kindt-Larsen, L., Kirkegaard, E., and Dalskov, J. 2011. Fully documented fishery: a tool to support a catch quota management system. *ICES Journal of Marine Science*, 68: 1606–1610.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. 2012. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems*, 2: 1097–1105.
- Larsen, R., Olafsdottir, H., and Ersbøll, B. K. 2009. Shape and texture based classification of fish species. *In* *Scandinavian Conference on Image Analysis*, pp. 745–749. Springer, Berlin, Heidelberg.
- LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86: 2278–2324.
- Li, X., Shang, M., Hao, J., and Yang, Z. 2016. Accelerating fish detection and recognition by sharing CNNs with objectness learning. *In* *OCEANS 2016-Shanghai*, pp. 1–5. IEEE.
- Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. 2017. Feature pyramid networks for object detection. *In* *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125.
- Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Zitnick, C. L. *et al.* 2014. Microsoft coco: common objects in context. *In* *European Conference on Computer Vision*, pp. 740–755. Springer, Cham.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., and Berg, A. C. 2016. Ssd: single shot multibox detector. *In* *European Conference on Computer Vision*, pp. 21–37. Springer, Cham.
- Long, J., Shelhamer, E., and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. *In* *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440.
- Lu, Y. C., Tung, C., and Kuo, Y. F. 2019. Identifying the species of harvested tuna and billfish using deep convolutional neural networks. *ICES Journal of Marine Science*, 77: 1318–1329.
- Manning, C. D., Manning, C. D., and Schütze, H. 1999. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA.
- Monkman, G. G., Hyder, K., Kaiser, M. J., and Vidal, F. P. 2019. Using machine vision to estimate fish length from images using regional convolutional neural networks. *Methods in Ecology and Evolution*, 10: 2045–2056.
- Morais, E. F., Campos, M. F. M., Padua, F. L., and Carceroni, R. L. 2005. Particle filter-based predictive tracking for robust fish counting. *In* *XVIII Brazilian Symposium on Computer Graphics and Image Processing (SIBGRAPI'05)*, pp. 367–374. IEEE.
- Needle, C. L., Dinsdale, R., Buch, T. B., Catarino, R. M., Drewery, J., and Butler, N. 2015. Scottish science applications of remote electronic monitoring. *ICES Journal of Marine Science*, 72: 1214–1229.
- Neubeck, A., and Van Gool, L. 2006. Efficient non-maximum suppression. *In* *18th International Conference on Pattern Recognition (ICPR'06)*, 3, pp. 850–855. IEEE.

- Qin, H., Li, X., Liang, J., Peng, Y., and Zhang, C. 2016. DeepFish: accurate underwater live fish recognition with a deep architecture. *Neurocomputing*, 187: 49–58.
- Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. 2016. You only look once: unified, real-time object detection. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788.
- Ren, S., He, K., Girshick, R., and Sun, J. 2015. Faster R-CNN: towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, 2015: 91–99.
- Russell, B. C., Torralba, A., Murphy, K. P., and Freeman, W. T. 2008. LabelMe: a database and web-based tool for image annotation. *International Journal of Computer Vision*, 77: 157–173.
- Shafry, M. R. M., Rehman, A., Kumoi, R., Abdullah, N., and Saba, T. 2012. FiLeDI framework for measuring fish length from digital images. *International Journal of Physical Sciences*, 7: 607–618.
- Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv : 1409.1556*.
- Spampinato, C., Chen-Burger, Y. H., Nadarajan, G., and Fisher, R. B. 2008. Detecting, tracking and counting fish in low quality unconstrained underwater videos. *VISAPP*, 2008: 1.
- Sung, M., Yu, S. C., and Girdhar, Y. 2017. Vision based real-time fish detection using convolutional neural network. *In OCEANS 2017-Aberdeen*, pp. 1–6. IEEE.
- Toh, Y. H., Ng, T. M., and Liew, B. K. 2009. Automated fish counting using image processing. *In 2009 International Conference on Computational Intelligence and Software Engineering*, pp. 1–5. IEEE.
- Tseng, C. H., Hsieh, C. L., and Kuo, Y. F. 2020. Automatic measurement of the body length of harvested fish using convolutional neural networks. *Biosystems Engineering*, 189: 36–47.
- Van Helmond, A. T., Chen, C., and Poos, J. J. 2017. Using electronic monitoring to record catches of sole (*Solea solea*) in a bottom trawl fishery. *ICES Journal of Marine Science*, 74: 1421–1427.
- Van Rossum, G., and Drake, F. L. Jr 1995. *Python Tutorial*. Centrum voor Wiskunde en Informatica, Amsterdam. 130 pp.
- White, D. J., Svellingen, C., and Strachan, N. J. 2006. Automated measurement of species and length of fish by computer vision. *Fisheries Research*, 80: 203–210.
- Zhuang, P., Xing, L., Liu, Y., Guo, S., and Qiao, Y. 2017. Marine animal detection and recognition with advanced deep learning models. *In CLEF (Working Notes)*.
- Zheng, Z., Guo, C., Zheng, X., Yu, Z., Wang, W., Zheng, H., Zheng, B., *et al.* 2018. Fish recognition from a vessel camera using deep convolutional neural network and data augmentation. *In 2018 OCEANS-MTS/IEEE Kobe Techno-Oceans (OTO)*, pp. 1–5. IEEE.


Handling editor: Cigdem Beyan



Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Original Article

Automated classification of schools of the silver cyprinid *Rastrineobola argentea* in Lake Victoria acoustic survey data using random forests

Roland Proud ^{1*}, Richard Mangeni-Sande^{1,2}, Robert J. Kayanda³, Martin J. Cox^{1,4}, Chrisphine Nyamweya⁵, Collins Ongore^{1,5}, Vianny Natugonza², Inigo Everson^{1,6}, Mboni Elison⁷, Laura Hobbs^{8,9}, Benedicto Boniphace Kashindye⁷, Enock W. Mlaponi⁷, Anthony Taabu-Munyaho^{2,3}, Venny M. Mwainge⁵, Esther Kagoya², Antonio Pegado¹⁰, Evarist Nduwayesu², and Andrew S. Brierley¹

¹*Pelagic Ecology Research Group, School of Biology, Scottish Oceans Institute, Gatty Marine Laboratory, University of St Andrews, St Andrews KY16 8LB, UK*

²*National Fisheries Resources Research Institute (NaFiRRI), PO Box 343, Jinja, Uganda*

³*Lake Victoria Fisheries Organization (LVFO), PO Box 1625, Jinja, Uganda*

⁴*Australian Antarctic Division, 203 Channel Highway, Kingston, Tasmania 7050, Australia*

⁵*Kenya Marine and Fisheries Research Institute (KMFRI), Mombasa, Kenya*

⁶*School of Environmental Sciences, University of East Anglia, Norwich Research Park, Norwich NR4 7TJ, UK*

⁷*Tanzania Fisheries Research Institute (TaFiRI), PO Box 475, Mwanza, Tanzania*

⁸*Scottish Association for Marine Science, Oban, Argyll PA37 1QA, UK*

⁹*Department of Mathematics and Statistics, University of Strathclyde, Glasgow G1 1XH, UK*

¹⁰*Instituto de Investigacao Pesqueira (IIP), Maputo, Mozambique*

*Corresponding author: tel: +44 (0)1334 46 3401; e-mail: rp43@st-andrews.ac.uk.

Proud, R., Mangeni-Sande, R., Kayanda, R. J., Cox, M. J., Nyamweya, C., Ongore, C., Natugonza, V., Everson, I., Elison, M., Hobbs, L., Kashindye, B. B., Mlaponi, E. W., Taabu-Munyaho, A., Mwainge, V. M., Kagoya, E., Pegado, A., Nduwayesu, E., and Brierley, A. S. Automated classification of schools of the silver cyprinid *Rastrineobola argentea* in Lake Victoria acoustic survey data using random forests. – ICES Journal of Marine Science, 77: 1379–1390.

Received 7 July 2019; revised 5 March 2019; accepted 10 March 2020; advance access publication 9 May 2020.

Biomass of the schooling fish *Rastrineobola argentea* (dagaa) is presently estimated in Lake Victoria by acoustic survey following the simple “rule” that dagaa is the source of most echo energy returned from the top third of the water column. Dagaa have, however, been caught in the bottom two-thirds, and other species occur towards the surface: a more robust discrimination technique is required. We explored the utility of a school-based random forest (RF) classifier applied to 120 kHz data from a lake-wide survey. Dagaa schools were first identified manually using expert opinion informed by fishing. These schools contained a lake-wide biomass of 0.68 million tonnes (MT). Only 43.4% of identified dagaa schools occurred in the top third of the water column, and 37.3% of all schools in the bottom two-thirds were classified as dagaa.

© International Council for the Exploration of the Sea 2020.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted reuse, distribution, and reproduction in any medium, provided the original work is properly cited.

School metrics (e.g. length, echo energy) for 49 081 manually classified dagaa and non-dagaa schools were used to build an RF school classifier. The best RF model had a classification test accuracy of 85.4%, driven largely by school length, and yielded a biomass of 0.71 MT, only c. 4% different from the manual estimate. The RF classifier offers an efficient method to generate a consistent dagaa biomass time series.

Keywords: artificial intelligence, big data, dagaa, Lake Victoria, machine learning, *Rastrineobola argentea*, school analysis, species identification, stock assessment

Introduction

In recent years, and parallel to the development of ever-cheaper computer-processor power, machine learning and artificial intelligence (AI) methods have been applied increasingly in ecology to ask “big questions” of “big data”. These methods have delivered promising results in species identification, biodiversity mapping and animal behaviour studies (Christin *et al.*, 2019). Active acoustic data collected during fish stock assessment surveys are a form of big data. A typical month-long survey can gather tens of gigabytes of data per narrowband frequency and, with the increasing inclusion of broadband echosounders and multibeam sonars in fish stock assessment surveys, this will increase by at least tenfold per vessel in the future (Demer *et al.*, 2017). Multiple autonomous platforms including wave gliders (Bingham *et al.*, 2012; Greene *et al.*, 2014) and saildrones (Mordy *et al.*, 2017; De Robertis *et al.*, 2019) are increasing the temporal and spatial coverage of fish stock monitoring, and the volume of data now being collected in some ecosystems exceeds institutional capacity for manual processing. Machine learning methods can potentially be utilized to automate data analysis pathways and, at the same time, reduce human error-induced uncertainty in stock biomass estimates.

During the analysis of acoustic survey data, visual scrutinization—classification by eye of features on echograms [the two-dimensional (2D) plots showing echo energy by depth and distance/time along track]—is often used to partition echo energy between species, but results can be operator dependent. Efforts to overcome this by the application of rigid “rules” can also be unsatisfactory. Identification of Antarctic krill (*Euphausia superba*), for example has been achieved by a simple “dB difference” approach that uses the difference in backscattering intensity between two frequencies as a diagnostic characteristic (Madureira *et al.*, 1993) but has in some areas of continental shelf been susceptible to erroneous inclusion of echoes from ice fish (Channichthyidae; Fallon *et al.*, 2016). In Lake Victoria, the silver cyprinid (*Rastrineobola argentea*; known locally as “dagaa”) is identified using a simple depth distribution rule that holds that most of the backscattered echo energy from the top third of the water column is from dagaa (LVFO, 2006). This approach—that was incorporated in to the Lake Victoria acoustic analysis standard operating procedure (SOP; LVFO, 2012) in a period when limited resources precluded anything more sophisticated—is, however, known to be flawed: not all the fish obey the rule. Here, we apply AI to the identification of echoes from schools of Lake Victoria fish in an effort to illustrate an example of the potential for AI in fisheries ecology and to improve the accuracy of stock assessments for the lake.

The Lake Victoria fishery

Lake Victoria is the world’s largest tropical lake (68 800 km²). Fisheries are vital for local food provision and for export earnings and contribute 2–3% to the gross domestic products of

the lake’s three riparian states (Uganda, Kenya, and Tanzania). Sustainable fisheries management is a regional priority, and there is an aspiration to move towards ecosystem-based fisheries management (LVFO, 2018).

Dagaa are a small (maximum length c. 9 cm) pelagic zooplanktivorous fish (Wanink, 1999) native to Lakes Victoria, Nabugabo, and Kyoga in East Africa. It is one of the few species in Lake Victoria to remain abundant following the introduction of the Nile Perch (*Lates niloticus*) in the 1950s (Goudswaard *et al.*, 2008; Sharpe and Chapman, 2014). Dagaa make up ~60% of the total annual Lake Victoria catch, with ~0.6 million tonnes (MT) being landed in 2015 (Mangeni-Sande *et al.*, 2019). Typically 100–500 kg of dagaa can be caught per boat per night using small seine nets and light attraction (LVFO, 2016a), and ~18 700 boats target dagaa (LVFO, 2016b). Dagaa fishing employs ~70 500 fishermen, and ~16 500 women are engaged in the labour-intensive drying of the catch (Okedi, 1981; LVFO, 2016b); fish are spread out for drying—often simply on the sand—in the sun and are turned regularly by hand. Dagaa are sold into the local and regional markets and consumed almost exclusively in southern and eastern Africa: dagaa is a cheap source of animal protein for the rural poor. High-quality dried fish are sold for human consumption, and lower-quality products (~70% of the total catch) are used for animal feed (Odongkara *et al.*, 2016).

The emphasis of research on fisheries in Lake Victoria has to date been largely on Nile Perch because of its importance in generating foreign currency revenue (US\$300 million; LVFO, 2018). However, for economic and ecological reasons, it is essential to establish effective management for sustainable exploitation of other species as well, including dagaa (Kolding *et al.*, 2019), and accurate estimates of stock biomass are an essential prerequisite for that.

Present estimates of dagaa biomass

Estimates of dagaa stock biomass are determined from acoustic data collected during bi-annual lake-wide fish stock assessment surveys. Dagaa, which are superficially similar to anchovy, are an obligate schooling pelagic species that possess swim bladders: as such dagaa is highly suitable for acoustic assessment. During daylight, dagaa aggregate into small schools (a few metres in length and height) that appear as distinct needle-like features in echograms when observed at typical survey setting, i.e. vessel speed of between 8 and 10 knots and ping intervals of between 0.2 and 0.5 s (Getabu *et al.*, 2003). Dagaa are presently evaluated by echo integration of 120 kHz data from the top one-third of the water column. It is assumed that all echo energy remaining in the top third of the water column after single-target detections (which are all attributed to Nile Perch; Kayanda *et al.*, 2012) have been removed arises from dagaa, and only dagaa. It is clear though, even from just a cursory reference to Getabu *et al.* (2003), that this is a false assumption: dagaa occupy a broader depth range than just the top third, and other species are known to inhabit

the top third. A new method for dagaa identification is needed urgently to improve the accuracy of stock assessment and, eventually, to improve the management that stems from the biomass estimate. Since the objective of the acoustic survey is to allocate all energy correctly, improving dagaa allocation will lead to improvements in the assessment of other species as well.

Fish school analysis using acoustic data

In order to establish reliable and reproducible methods to identify and discriminate species detected acoustically during surveys, we need first to identify acoustic characteristics, or sets of characteristics, that are unique to particular target species and that are therefore diagnostic. For schooling species, these characteristics can be at the school level (rather than at the level of the individual fish), and the physical shape, echo intensity, frequency response and behaviour of schools of different fish species can be diagnostic (Coetzee, 2000; Reid *et al.*, 2000; Lawson, 2001; Bertrand *et al.*, 2008; Fernandes, 2009; Paramo *et al.*, 2010). Since the development of standard methods for extracting school characteristics (Barange, 1994; Coetzee, 2000; Reid *et al.*, 2000; Diner, 2001), analyses have been conducted to study the shapes and behaviours of schools of many species of fish (Lawson, 2001; Fernandes, 2009; Fallon *et al.*, 2016) and the swarm characteristics of krill (Tarling *et al.*, 2001; Klevjer *et al.*, 2010; Cox *et al.*, 2011). Such analyses are now being used to aid species identification, and hence to reduce uncertainty around estimates of fish stock biomass (e.g. for herring and mackerel; Fernandes, 2009).

Schools of a specified minimum size (horizontal and vertical dimensions) and echo intensity can be extracted automatically from acoustic observations [both 2D observations from conventional vertical echosounders and three-dimensional (3D) observations from multibeam sonar surveys], and school metrics pertaining to morphology, position, and acoustic scattering properties (e.g. echo energy across different frequencies) can be collated to characterize schools (Barange, 1994). Performing such an automated school extraction process for a typical month-long Lake Victoria vertical echosounder survey results in over 100 000 extracted schools. These include schools of dagaa, small (<10 cm) Nile Perch, and haplochromine cichlids (aggregations of the pelagic crustacean *Caridina nilotica* are also apparent). More than 25 acoustic surveys have been conducted on Lake Victoria over the past 20 years (Taabu-Munyaho *et al.*, 2014), and school data within them offer an incredibly valuable resource for examining potential change as a function of, for example, fishing pressure and environmental variability (Brierley and Cox, 2010, 2015). Fundamental to these types of analyses and indeed to fish stock assessment are consistent and reproducible methods to identify and discriminate species, including dagaa. It is impractical to attempt to use manual visual scrutinization to discriminate dagaa schools from the more than 2.5 million estimated schools now potentially accessible from the combined 25-survey database. Therefore, the main objective of the work reported here was to develop a robust and consistent approach that was both cost- and time-effective, and that used machine learning/AI to perform the automatic classification of dagaa schools (e.g. Fernandes, 2009; Cox *et al.*, 2011; Fallon *et al.*, 2016; Escobar-Flores *et al.*, 2019).

Machine learning

It is now common practice to use machine learning techniques to classify data (Malde *et al.*, 2019). Features isolated in acoustic

survey data, such as schools, scattering layers, and single targets, have been classified using a wide range of machine learning techniques including mixture models (Fleischman and Burwen, 2003; Escobar-Flores *et al.*, 2018), artificial and convolutional neural networks (Haralabous and Georgakarakos, 1996; Simmonds *et al.*, 1996; Korneliussen *et al.*, 2016; Brautaset *et al.*, 2020), decision trees, random forests (RFs), and boosted regression trees (Fernandes, 2009; D'Elia *et al.*, 2014; Fallon *et al.*, 2016; Escobar-Flores *et al.*, 2018, 2019), discriminant-function analysis and principal components analysis (Nero and Magnuson, 1989; Scalabrin *et al.*, 1996; Brierley *et al.*, 1998; Lawson, 2001), and k-means clustering (Tegowski *et al.*, 2003; Proud *et al.*, 2017). Ensemble tree methods (e.g. RF and boosted regression trees) have only been adopted in the past decade but have been found to be particularly good (having high accuracy) for classifying fish schools (Fernandes, 2009; D'Elia *et al.*, 2014; Fallon *et al.*, 2016).

Objective of the present study

The objective of this study is to develop a robust, automated method to identify echoes from dagaa schools in echosounder data collected during Lake Victoria fish stock assessment surveys. Previous work (Getabu *et al.*, 2003; LVFO, 2006), and a large accumulation of local experience, suggests that dagaa form schools that have a distinct needle-shaped (vertically tall, horizontally narrow) appearance in underway echograms. We set out first to confirm that needle-shaped acoustic features are in fact dagaa schools, and then to develop a machine learning method to identify dagaa schools amongst all extracted schools. In this study, we make no attempt to classify aggregations of the other common Lake Victoria pelagic species because there is presently not enough ground-truth data (e.g. trawl data) to underpin such an analysis.

Methods

Determining the characteristics of dagaa schools

We conducted target fishing during an October 2019 field study in a coastal region (c. 40 m lakebed depth) of the Ugandan sector of the lake. We fitted a standard Lake Victoria bottom trawl with a fine-mesh cod-end cover and used this to target needle-like and non-needle-like pelagic echogram features. The net had an estimated vertical opening of <10 m, and was fished at 4 knots for 15 min at each sampled depth. Catch samples were sorted into species groups and the individuals in each group were counted, measured and weighed. The acoustic data recorded during each trawl were resampled to typical survey settings (vessel speed = 9 knots and ping interval = 0.2 s) to reconstruct echograms that would have been produced had the fished schools been encountered at typical survey speed.

Acoustic survey data collection

Acoustic and environmental data collected during the November 2015 fish stock assessment survey in Lake Victoria (LVFO, 2015) were used to build a dagaa school classifier. That survey was selected because, when this work began, it was the most recent survey that had been processed. The survey was conducted from research vessel (RV) *Victoria Explorer* between 1 and 29 November (including 4 days breaks for re-provisioning). It covered c. 4 000 km of survey track over most of the lake, across a range of lakebed depths between 2 and 70 m. Most of the survey was conducted in daylight hours, and sampling effort was highest

in the more productive inshore regions of the lake (Figure 1). The night-time vertical distribution and echogram appearance of dagaa schools may differ from the daytime distribution and form, and so night-time observations were excluded from the analysis: only acoustic data collected between sunrise and sunset (excluding astronomical twilight) were analysed. Acoustic data were collected using two hull-mounted Kongsberg (Horten, Norway) Simrad EK60 scientific echosounders operating at 70 and 120 kHz, both with a 7° nominal beam width. A pulse length of 0.256 ms was used, with a ping interval of 0.2 s. A standard split-beam echosounder calibration (Foote *et al.*, 1983; Demer *et al.*, 2015) was carried out prior to the survey. In this study, we use and report only 120 kHz data because one objective of the work is to develop a route for the reanalysis of historic surveys, and early surveys only used 120 kHz. However, at an early stage of this study, we investigated the benefit of including 70 kHz data as well but found no improvement in decision tree-based school classification using two frequencies.

Hydrographic measurements were taken at predefined stations ($N=58$, see Figure 1) using a Sea and Sun Conductivity, Temperature, and Depth (CTD) probe and a YSI 650 multi-parameter sonde to measure temperature (°C), dissolved oxygen concentration (DO, mg l^{-1}), conductivity ($\mu\text{S cm}^{-1}$), pH, turbidity [Formazin Turbidity Units (FTU)], and chlorophyll *a* concentration ($\mu\text{g l}^{-1}$).

School extraction and manual classification

The “Schools Detection Module” in Echoview software (v9; Myriax, Hobart, Tasmania) was used to extract all schools from the echosounder data. Before running the school detection algorithm, echosounder data were thresholded at $-54 \text{ dB re } 1 \text{ m}^{-1}$ (i.e. any samples below this value were excluded from analysis).

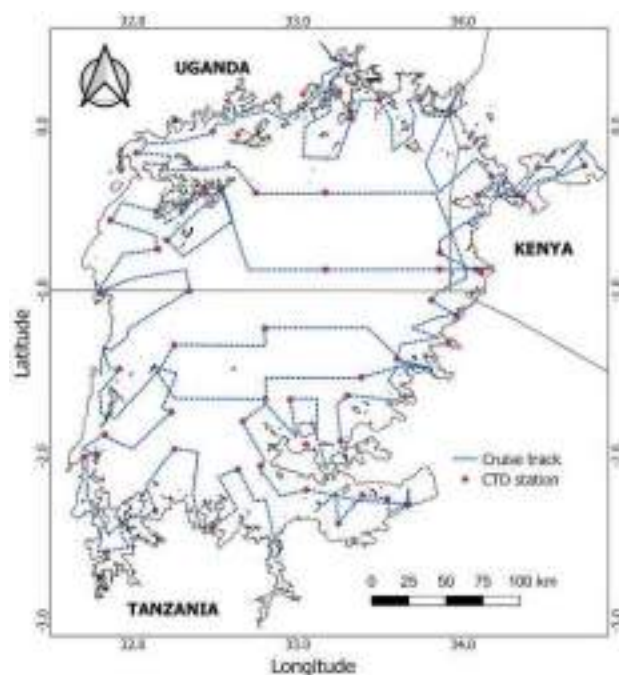


Figure 1. Map of Lake Victoria, East Africa, showing the cruise track and the CTD stations (solid points) for the November 2015 fish stock assessment survey.

Recalling that $S_v = 10 \times \log_{10}(10^{(TS/10)} \times \text{packing density})$, the threshold was set with the consideration of an expected mean target strength (TS) at 120 kHz of dagaa with a mean length of 5.3 cm of $-57.6 \text{ dB re } 1 \text{ m}^2$ and a very conservative minimum school packing density of c. 2 fish per m^3 (Tumwebaze, 2003).

The school detection algorithm [Shoal Analysis and Patch Estimation System (SHAPES)] is based on the work of Barange (1994) and Coetzee (2000) and requires a number of parameters to be set. From preliminary analysis, local prior knowledge, and the work of Getabu *et al.* (2003), dagaa schools were perceived to be characteristically very narrow (a few pings) relative to vertical extent (tens of samples) in echograms, dense, and compact (i.e. without any vacuoles or holes), and so the SHAPES algorithm parameters were set conservatively to ensure that all schools of this nature, as well as schools with the more usual rounded echogram appearance, would be captured. Thus, all school detection parameters, except for the horizontal linking distance, were set to their minimum possible values, i.e. the minimum candidate height, minimum candidate length, minimum school height, minimum school length, and vertical linking distance were all set to 1 m. The maximum horizontal linking distance was set to 5 m to ensure that, given the vessel speed and ping rate, consecutive pings could be linked. The SHAPES algorithm was run across the entire acoustic dataset. It identified schools with a diversity of forms, from small compact needle-like schools typical of dagaa (see Figure 2) to large amorphous schools hundreds of metre in length that were layer like in appearance. The expert view is that

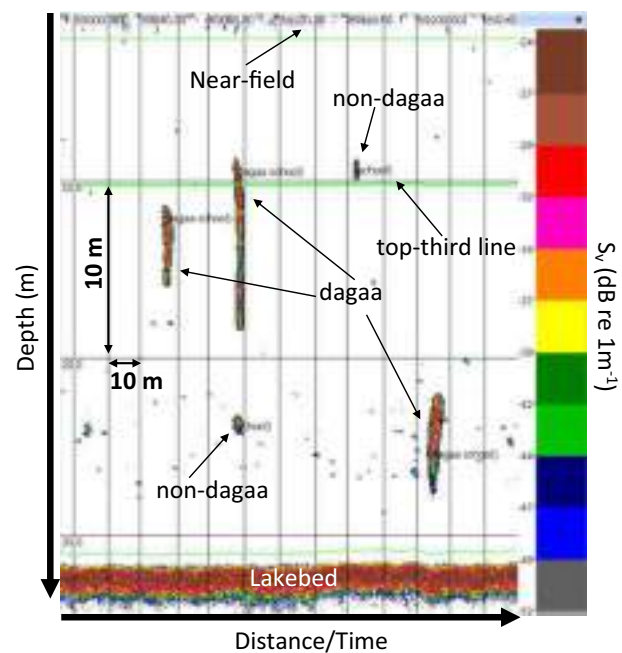


Figure 2. Echoview-generated 120 kHz echogram of automatically detected and manually labelled schools (grid size 10 by 10 m). Three dagaa schools are labelled and have the characteristic needle-like echogram appearance (vessel speed = 9 knots and ping interval = 0.2 s). In this example, the lakebed is at 32 m and the top third line (that under the existing standard operating procedure would demark the lower limit of the dagaa habitat) is at 10.7 m: the dagaa schools extend deeper than the top third line and the non-dagaa schools can be seen above the top third line, well illustrating the inability of depth alone to differentiate dagaa and non-dagaa.

these layers are comprised of haplochromine cichlids and small Nile Perch (<10 cm). Any “schools” that were longer than 100 m in length were deemed to be scattering layers (Proud *et al.*, 2015) and were excluded from further analysis. All remaining schools were examined manually and categorized by eye (visual scrutiny) as either dagaa or non-dagaa.

School metrics (Table 1) were exported from Echoview. Environmental variables (Table 1) were ascribed to each school as those at the nearest CTD station by distance.

Using machine learning to build a school-based classifier

An RF model was built using a subset of the manually identified schools. During the survey, on-transect vessel speed varied between 6 and 13 knots, but c. 84% of effort was between 8 and 10 knots (which is the typical range of survey speeds across all historic acoustic surveys on the lake). Historic manual classification has identified needle-like echo traces as dagaa schools, but the aspect ratio (height to width) of echogram features is of course a function of vessel speed and ping rate. This raises the possibility that schools detected at slow speed would be rejected by eye as dagaa because they would appear too rounded: this is an important illustration of one of the weaknesses of visual classification methods. To avoid incorporating any potential speed-related bias in the RF model of dagaa school characteristics, only visually classified schools detected in the range of usual survey speeds (8–10 knots) were used to build the RF model.

Following the standard RF protocol (Breiman, 2001), schools remaining after speed filtering were split randomly into a training dataset (80% of data are typically used to train an RF classifier, and we adhered to that) and a test dataset (20% of data are typically used to test RF classifiers). The R packages “caret”, “party”, and “trees” (Strobl *et al.*, 2008, 2009; Kuhn, 2019; Ripley, 2019) were used to build RF models. RF algorithms have two tuning parameters: these are *mtry*, the number of variables to select randomly from the total available list of school metrics (Table 1) when splitting data at each node in a tree, and *ntrees*, the number of trees to build. In this study, *mtry* was initially set to 4 and *ntrees* to 500 (these are the default values), but a range of different

mtry and *ntrees* values was also used to assess their impact on RF classification accuracy.

We used repeated (three times) tenfold cross-validation to assess the accuracy of the RF (Stone, 1974; Breiman, 2001). This validation process involved splitting the training dataset into ten equally sized subsets (or folds), building the RF model using a dataset containing nine of the tenfolds, and then validating the model on the other remaining fold. This process was repeated ten times such that each fold acted as the validation dataset once. This process was repeated three times (with random, so probably different, tenfold splitting on each of the three occasions), and the accuracy of the model was calculated by taking an average over the resultant 30 accuracy values (3×10 folds).

Assessment of the RF model

The RF model was assessed using the mean and standard deviation of the training accuracy, and the kappa statistic κ (the proportion of classification agreement beyond that expected to occur by chance, where $\kappa = 0$ is suggestive of classification only matching what would be expected by random chance assuming a binomial distribution; Cohen, 1960). RF models are difficult to interpret, since they are typically comprised of hundreds of fully grown decision trees. In the majority of cases, RF models are assessed by accuracy metrics and the importance of each predictor (each school metric in this case) is assessed by single specific or multiple so-called “importance metrics” (Breiman, 2001). Here, we use conditional variable importance (Strobl *et al.*, 2008) to assess each predictor’s ability to discriminate between target classes (i.e. dagaa or non-dagaa): unlike other importance measures (e.g. mean decrease in accuracy), conditional variable importance is robust against correlated variables (Fallon *et al.*, 2016), e.g. water temperature and school depth are likely to be correlated.

Dagaa stock biomass estimates

The RF model, which was built using a subset of the extracted schools, was used to classify the entire dataset of schools from the 2015 survey. School-based estimates of dagaa stock biomass were then calculated using both the manually classified schools and the

Table 1. School metrics used to build an RF model to classify detected schools

School metric	Description	Unit
Length	Mean length of school corrected for beam width	m
Depth	Mean depth of school	m
Height	Mean height of school corrected for pulse length	m
Image compactness	School perimeter squared/ $(4 \times \pi \times$ school area); for a perfectly circular school this would be 1	Unitless
NASC	School NASC is an historic acoustic unit that is the average amount of echo energy produced by the school per m^2 of lake surface, scaled up to an area of 1 nautical mile squared	$\text{m}^2 \text{nmi}^{-2}$
Lakebed depth	Depth of lakebed as detected by the 120 kHz echosounder	m
Temperature	Measured value at school depth obtained from the closest CTD station	$^{\circ}\text{C}$
DO	Measured value at school depth obtained from the closest CTD station	mg l^{-1}
pH	Measured value at school depth obtained from the closest CTD station	Unitless
Turbidity	Measured in Formazin Turbidity Units. Measured value at school depth obtained from the closest CTD station	FTU
Chlorophyll <i>a</i> concentration	Measured value at school depth obtained from the closest CTD station	$\mu\text{g l}^{-1}$
Longitude	Taken from vessel GPS	Degrees East
Latitude	Taken from vessel GPS	Degrees North
Time of day	Decimal time, calculated from vessel GPS	Hours

RF classified schools. Echo energy from schools classified as dagaa (either manually or via the RF model) was converted into biomass following the Lake Victoria Fisheries Organization SOP for stock assessment (LVFO, 2012). Accordingly, mean dagaa nautical area scattering coefficient (NASC) values were determined for each of the 18 SOP-defined lake areas, which are split by country (3; Uganda, Tanzania, and Kenya), lake quadrant (4; NW, NE, SE, SW), and depth (3; “inshore” <10 m; “coastal” 10–40 m, and “deep” >40 m). These 18 mean NASC values were converted to biomass density ($T\ m^{-2}$) using the mean dagaa TS per kg (TS_{kg} , i.e. the amount of 120 kHz echo energy produced by 1 kg of dagaa) of $-29.4\ dB\ kg^{-1}$ (Tumwebaze, 2003). Biomass densities were multiplied by associated areas to scale to biomass (T) in each of the 18 areas, and these were summed to give a whole-lake value. This process was repeated 1000 times, resampling with

replacement dagaa school NASC values by area on each iteration (i.e. bootstrapping), and 95% confidence intervals were calculated.

Results

A total of 120 181 schools (larger than 1 m in length and height) were detected by the SHAPES algorithm in the echosounder data. Schools in “bad data” regions (e.g. sections of transect with no GPS) and schools detected at night were removed reducing the useable dataset to 115 778 schools.

Confirmation that needle-like echo traces are dagaa schools

It is generally believed, based on the work of Getabu *et al.* (2003) and on accumulated local expert opinion, that schools with a needle-like appearance in 120 kHz underway echograms are dagaa schools. To this end, needle-like schools were fished during an October 2019 field study (Figure 3 and Table 2).

A total of 93 schools were detected acoustically during Haul 1 (near surface), and 99.5% of the total catch by number (399 fish) was dagaa. The only non-dagaa component of the combined catch was two haplochromine cichlids, each just 4 cm long (the mean length of dagaa was c. 3.8 cm). These cichlids would have contributed c. 1% to integrated trawl echo energy (estimated using haplochromine $TS = 20\log L - 66.65$; LVFO, 2015). Dagaa and needle-like schools were also present in Hauls 2–5 along with similar numbers of haplochromines, conforming with the view of Getabu *et al.* (2003) that dagaa are not restricted to the near-surface layer (Table 2). However, since catch obtained from Hauls 2–5 was likely contaminated during time spent at the surface whilst deploying and recovering the net, these observations were not quantitatively assessed.

Manual classification of schools using the lake-wide 2015 survey data

A total of 56 079 of the 115 778 schools passed for manual visual identification were classified as dagaa. The remaining 59 699 schools, judged by experts to be non-dagaa, would have contained haplochromines, *Tilapia* spp., small Nile Perch (<10 cm) and other species, but the present state of knowledge is

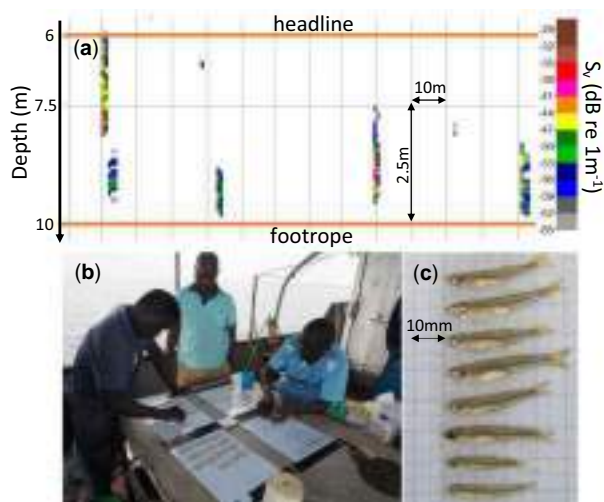


Figure 3. Example dagaa trawl (Haul 1) during the October 2019 field study in Ugandan waters aboard the RV *Ibis*: (a) 120 kHz echogram showing needle-like schools, which are commonly believed to be dagaa schools; (b) catch from needle-like schools being sorted; and (c) dagaa, which comprised >99% of the catch by number.

Table 2. Net haul and catch information (numbers of individual fish)

Haul	Wire out (m)	Headline depth (m)	Dagaa (N)	Haplochromines (N)	Needle-like schools (N)
1	25	6	399	2	93
2	50	11	288	1	16
3	75	26	73	86	0
4	100	30	86	103	0
5	125	34	68	22 ^a	0

The water depth was 40 m.

^aThe decapod *Caridina nilotica* was also present, in small number, in the catch from Haul 5.

Table 3. Distributions of dagaa and non-dagaa schools by depth according to expert manual classification, and median lake-wide biomass estimates (bootstrapped 95% confidence intervals given in square brackets)

Depth zone	Dagaa schools (N)	Non-dagaa schools (N)	Total schools (N)	Dagaa school biomass (T)
Top third	24 357	6 403	30 760	370 701 [361 388–379 408]
Bottom two-thirds	31 722	53 296	85 018	312 707 [301 747–324 400]
Total	56 079	59 699	115 778	683 107 [668 957–697 721]

Table 4. Distributions of daga and non-daga schools by depth according to RF classification, and median lake-wide biomass estimates (bootstrapped 95% confidence intervals given in square brackets)

Depth zone	Daga schools (N)	Non-daga schools (N)	Daga school biomass (T)
Top third	26 171 (+7.44%)	4 589 (−28.33%)	394 373 [385 006–403 609] (+6.38%)
Bottom two-thirds	32 534 (+2.56%)	52 484 (−1.52%)	315 853 [305 435–327 424] (+1.01%)
Total	58 705 (+4.68%)	57 073 (−4.40%)	710 547 [695 426–725 205] (+4.02%)

Brackets indicate percentage change relative to manual classification.

insufficient to classify them by species; that will be the task for a subsequent project.

Only 43.4% of the manually classified daga schools occurred in the top third of the water column, but 89.3% of non-daga schools occurred in the bottom two-thirds: together these proportions give the “top third” method an overall school classification success rate by number of c. 72.6% (see Table 3). Daga school biomass was found to be almost equally distributed between the top third and bottom two-thirds of the water column. The daga stock biomass estimate arising from the manual classification was 0.68 MT (see Table 3).

RF model

The 49 081 manually classified schools remaining after filtering for vessel speed were split into a training dataset (13 547 daga schools, 25 718 non-daga schools) and a test dataset (3 319 daga, 6 497 non-daga). The training dataset was used to train the RF classifier. An RF classifier was constructed using all 14 available school and environment metrics (Table 1). The default values of *mtry* (4) and *ntrees* (500) produced the best model as evaluated by model accuracy; other *mtry* and *ntrees* parameter values were tested (*mtry*: 2–8 and *ntrees*: 200–2000) but provided no improvement in accuracy. The RF model had a training classification accuracy of 85.0% (*SD* = 0.49%), a test classification accuracy of 85.4%, and a κ -value of 0.66 (*SD* = 0.011).

RF predictions

The RF model was used to classify all schools in the full dataset of 115 778 schools (i.e. not just the schools that passed the speed filter). Since school dimensions were determined from GPS position, there were no speed-related artefacts in the automatically extracted school metric values. Schools classified by the RF model as daga were used to estimate lake-wide biomass, and are summarised in Table 4. The RF-derived biomass value differed by only 4.02% from the manual school classification result (see Tables 3 and 4). The largest difference between manual classification and the RF model classification was of non-daga schools in the top third depth zone. The manual scrutinization classified 1 814 more schools as non-daga. We believe that this occurred when slow vessel speed served to stretch observations of daga schools horizontally, giving them a non-daga appearance in the echogram. The RF approach takes school dimensions from GPS locations so is not “misled” by variability in vessel speed.

Importance of different school metrics to overall RF model effectiveness

Evaluating the importance of each school metric (Table 1) to the RF model, regardless of any correlation between the metrics (known as “conditional variable importance”), showed that

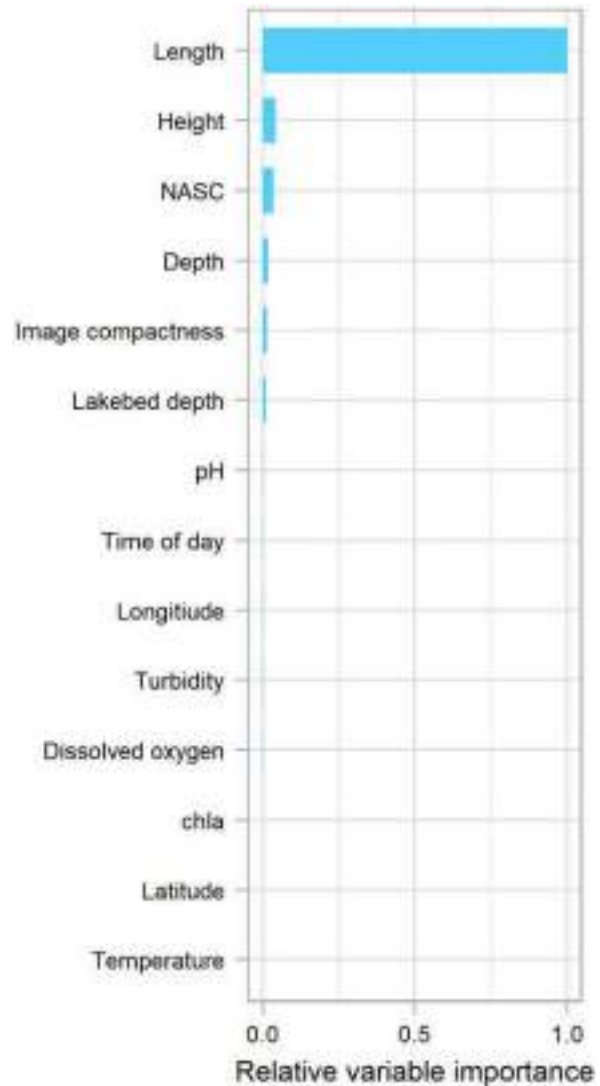


Figure 4. Relative variable importance (conditional variable importance normalized between 0 and 1) for school metrics used to build the RF model.

school length was the most important metric, followed by school height, school NASC, school depth, school image compactness, and lakebed depth (Figure 4). Environmental variables other than lakebed depth contributed very little to the overall predictive power of the model, and when all environmental information was removed, the overall RF accuracy reduced by only c. 1%. This suggests that, during the 2015 survey, school structure was not influenced strongly by environmental variability across Lake Victoria.

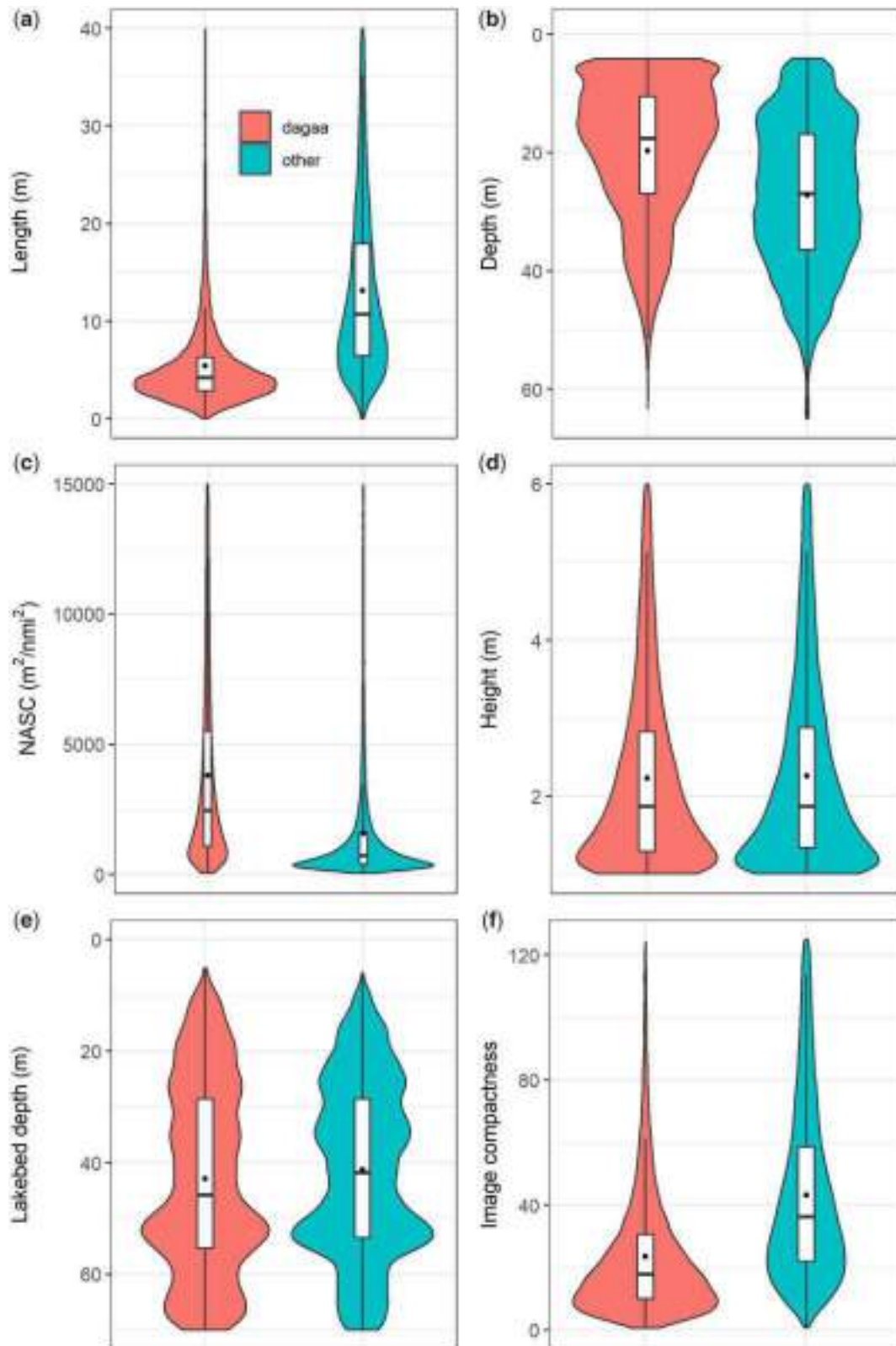


Figure 5. Violin plots, which show smoothed probability density, with boxplots overlaid, for important school metrics used in the RF model to classify dagaa schools. Plotted school metrics are: (a) school length, (b) school depth, (c) school nautical area scattering coefficient (NASC) value, (d) school height, (e) lakebed depth and (f) school image compactness. Black filled circles show distribution means.

School metrics

Distributions of the school metrics found to be important for dagaa classification were plotted as violin (Hintze and Nelson, 1998) and box plots, displaying the first quartile (Q1), median (M), third quartile (Q3), and probability density of each distribution (Figure 5). Dagaa school length (Figure 5a) (Q1 = 2.88 m; M = 4.23 m; Q3 = 6.34 m) was found to be significantly different to non-dagaa school length (Q1 = 6.82 m; M = 11.63 m; Q3 = 20.87 m; Kolmogorov-Smirnov (KS) test: $p < 0.001$; $D = 0.53$), a finding which provides quantitative support for the descriptive picture painted by Getabu *et al.* (2003) of dagaa schools as “needles”. Both dagaa and non-dagaa schools were found across all lake strata (inshore, coastal, and deep—see Figure 5b), but dagaa schools were typically found shallower in the water column (Q1 = 10.51 m; M = 17.63 m; Q3 = 26.94 m) than the non-dagaa schools (Q1 = 16.97 m; M = 27 m; Q3 = 36.48 m), which is some limited endorsement of the simple “top third” rule (but note that there are many dagaa schools in deeper water that the third-rule does not capture). School heights were similar between dagaa and non-dagaa schools (Figure 5d), but image compactness values of dagaa (Q1 = 10.02; M = 18.03; Q3 = 31.3) were significantly smaller (KS test: $p < 0.001$; $D = 0.38$) than non-dagaa (Q1 = 23.38; M = 40.29; Q3 = 72.19), i.e. in equidimensional x, y space, dagaa schools were paradoxically actually the more circle like in appearance: although appearing as needle-like features in echograms, if the aspect ratio of the image was to be set to 1:1, dagaa schools would in fact appear as squashed circles with a median length and height of 4.23 and 1.68 m, respectively (see Figure 5).

Discussion

We have developed a new automated and standardized method to classify schools extracted from Lake Victoria echosounder data as either dagaa or non-dagaa using an RF model. The RF model had a school classification accuracy of 85.4% as judged against a test dataset of 9 816 manually classified schools. When used to classify all detected schools, the RF model picked out schools that resulted in a total lake-wide biomass of c. 0.71 MT, which was within c. 4% of the biomass derived from schools classified manually as being dagaa (0.68 MT): bootstrapped confidence limits for biomasses arising from manual and RF classification overlapped.

Implications for fish stock management

The dagaa biomass estimate reported here of c. 0.7 MT is likely to be an underestimate for several reasons, such as: (i) since vertical echosounders are used to collect the data, and because dagaa are known to occupy shallow depths, some of the signal will be lost in the acoustic near-field (approximately the top 1.85 m for the 120 kHz transducer presently used); (ii) a component of the fish population may respond to the vessel (most likely avoiding, but possibly being attracted) (Brehmer *et al.*, 2019); and (iii) dagaa schools observed at present survey settings are relatively narrow (just a few pings in length) and it is possible that some particularly narrow schools (<1 m in length in some dimension) are not detected because the distance along survey track between consecutive pings, from beam edge to beam edge, is >school length. Although conservative, the biomass is determined by what will be a reproducible method that will be able to deliver an internally consistent relative index of variability over the years that will be valuable for management under the precautionary approach

(Francis, 1996). Suggestions for progressing towards absolute dagaa stock estimation are given below, and include the use of multibeam sonar to sample the near surface.

Method performance

The RF classifier provides a robust and consistent means of dagaa school classification that, assuming software capable of performing school extraction is available, is both time- and cost-effective: the RF approach can achieve in minutes a classification task that, for the November 2015 survey (classifying manually 120 181 schools), took c. 100 person hours. The RF method will enable repeatable estimates of dagaa stock biomass to be calculated (estimates that would not be subject to any potential expert operator bias) and make this component of the stock assessment process resilient to the loss of expertise that might arise due to changes in personnel. Assuming stability in school morphology over time (and there is evidence from stocks of other pelagic species that this is likely; Cox *et al.*, 2011; Brierley and Cox, 2015), the RF method will enable the reanalysis of historic data (there are ~20 pre 2015 surveys), and future surveys (surveys are accumulating at ~2 per year presently) in an equivalent manner to produce robust and consistent time series.

One of the strengths of the RF classifier is that it uses actual length/widths/echo energies of schools to identify them, rather than relying on a visual interpretation of a feature the appearance of which will be influenced by vessel speed, ping rate, colour scale, feature depth, and echosounder beam angle (see Diner, 2001). In recognition of these potential impediments to successful and reliable visual classification, the RF model was built using only schools detected at usual survey speeds (8–10 knots), so avoiding the distortion in school appearance at the extremes of vessel speed that we believe is at the root of the differences in numbers of schools classified as dagaa/non-dagaa by RF and manual methods. In future studies relying on visual identification to test AI approaches, prior to visual scrutiny, echosounder data should be resampled in distance such that ping width is constant and consistent with typical survey settings. Changes in school width with depth (as the acoustic beam widens) should also be accounted for (Diner, 2001).

Potential for future development

Vertically orientated echosounders commonly used in fish stock assessment (Fernandes *et al.*, 2002; Simmonds and MacLennan, 2005) have very narrow beam widths at short range (at 10 m range, the acoustic beam of the 120 kHz echosounder used in this study has a width of c. 1.2 m) and so offer a limited window of observation on species that inhabit the near surface. Consequently, the pelagic trawl used to fish near-surface dagaa schools in this study (Table 2 and Figure 3) would likely have encountered many schools that were not detected acoustically. Near-field effects also mean that echo returns from close to the transducer (c. 1.85 m in the case of the 120 kHz transducer used here) are not quantitatively reliable, such that typical surveys are effectively blind to the top few metres, potentially missing biomass. Use of multibeam sonar, instruments that typically sample a fan of acoustic beams spanning up to 180° beneath the vessel, or horizontally oriented echosounders, can open a window on the near surface (Gerlotto *et al.*, 1999; Paramo *et al.*, 2010). Multibeam has been used to make 3D measurements of fish schools at or close to the surface, and has also delivered valuable

data on the scale of avoidance by schools of research vessels (Gerlotto *et al.*, 2004). Incorporating multibeam instrumentation into Lake Victoria fish stock assessment surveys would effectively increase the volume of the lake sampled, provide valuable information with regard to school morphology, lead to more school detections for a given area (which could be readily integrated into the RF model) and hence reduce uncertainty in fish stock biomass estimates.

The RF model was trained and tested using data collected during a single survey (it was impractical to try to manually classify schools from more than one survey given available resources), but a future objective is to apply the RF classifier to the full range of available survey data (1997–present). We will need then to be wary of the potential for seasonal and/or annual changes in school characteristics. Lake Victoria shows strong seasonal physical change between fully mixed in the rainy season and stratified in the dry season. Deeper waters can become oxygen depleted in stratified times (Njiru *et al.*, 2012), and this may serve to vertically restrict dagaa habitat. Vertical habitat compression has been reported in the seas off Peru when the oxycline shallows (Bertrand *et al.*, 2008). Year-to-year variability in school structure may be less important: work on a variety of species over years spanning strong fluctuations in stock biomass has suggested that school shape does not vary significantly, but rather that it is the number of schools that varies with fluctuations in stock biomass (Brierley and Cox, 2015).

Between 2005 and 2014, total Lake Victoria fish stock biomass (including dagaa, Nile Perch, haplochromine cichlid species and others) has, on the basis of echosounder data analysis, appeared to be stable at c. 2.5 MT (Taabu-Munyaho *et al.*, 2016): the biomasses of Nile Perch and dagaa have both appeared to fluctuate from year to year, but in opposite directions. It is questionable how this apparent total biomass invariability can be ecologically possible given the greatly varying sizes, trophic levels, and ages-at-maturity of dagaa and Nile Perch. How much of this apparent zero-sum game is an artefact of fish not obeying the “top third” rule remains to be determined and will be the subject of an investigation that the repeatable RF classifier developed here will enable.

The next step will be to recalculate the time series of dagaa biomass from school information extracted from 20 years’ worth of acoustic survey data. This will be achieved by (i) pre-processing of the historic acoustic survey data (e.g. filtering noise spikes, which may resemble dagaa schools) and collating calibration results; (ii) building a new training dataset, composed of schools manually classified in different seasons and years, to study temporal changes in dagaa distribution, and investigate the validity of the “top third” method and drift in RF model parameters across the time series; (iii) applying geostatistical and or maximum entropy methods (Petitgas, 2001; Brierley *et al.*, 2003) to map dagaa echo intensity; and (iv) converting echo intensity to biomass using the latest measurements of dagaa *TS* and length–weight relationships derived from catch data. The new Lake Victoria dagaa biomass time series will enable any emerging interannual fluctuations in biomass to be considered in light of annual catches and environmental variability.

Concluding remarks

The work reported here is a first step in moving Lake Victoria fisheries data analysis towards a fully automated processing chain built on machine learning and AI methods. Due to the automated

nature of these methods, time-series reanalysis will no longer be impractical and a severe drain on resources, but will be achievable rapidly with minimal manual effort. This will pave the way for a spectrum of studies on spatial and temporal variability in species distributions and progress along the road to ecosystem-based management of Lake Victoria fisheries, and to underpinning sustainable economy and food security in East Africa (Kolding *et al.*, 2019).

Acknowledgements

We thank the participants of a workshop (Hydro-acoustic data analysis using R) funded by the University of St Andrews and the University of Strathclyde held at the Kenya Marine and Fisheries Research Institute (KMFRI) 17–21 June 2019 for insightful discussion of the RF method.

Funding

Acoustic assessments of fish stocks in Lake Victoria have been supported by multiple funders over the years. The dagaa classification reported here was supported specifically by several Scottish Funding Council Global Challenge Research Fund (GCRF) grants from the University of St Andrews and the University of Strathclyde, by a GCRF Networking Grant to ASB and RJK from the UK Academy of Medical Sciences (GCRFNG\100371), and a Royal Society International Collaboration Award to ASB and Rhoda Tumwebaze, LVFO (ICA\R1\180123).

References

- Barange, M. 1994. Acoustic identification, classification and structure of biological patchiness on the edge of the Agulhas Bank and its relation to frontal features. *South African Journal of Marine Science*, 14: 333–347.
- Bertrand, A., Gerlotto, F., Bertrand, S., Gutiérrez, M., Alza, L., Chipollini, A., Díaz, E., *et al.* 2008. Schooling behaviour and environmental forcing in relation to anchoveta distribution: an analysis across multiple spatial scales. *Progress in Oceanography*, 79: 264–277.
- Bingham, B., Kraus, N., Howe, B., Freitag, L., Ball, K., Koski, P., and Gallimore, E. 2012. Passive and active acoustics using an autonomous wave glider. *Journal of Field Robotics*, 29: 911–923.
- Brautaset, O., Waldeland, A. U., Johnsen, E., Malde, K., Eikvil, L., Salberg, A., and Handegard, N. O. 2020. Acoustic classification in multifrequency echosounder data using deep convolutional neural networks. *ICES Journal of Marine Science*, 77: 1391–1400.
- Brehmer, P., Sarré, A., Guennégan, Y., and Guillard, J. 2019. Vessel avoidance response: a complex tradeoff between fish multisensory integration and environmental variables. *Reviews in Fisheries Science & Aquaculture*, 27: 380–391.
- Breiman, L. 2001. Random forests. *Machine Learning*, 45: 5–32.
- Brierley, A. S., Gull, S. F., and Wafy, M. H. 2003. A Bayesian maximum entropy reconstruction of stock distribution and inference of stock density from line-transect acoustic-survey data. *ICES Journal of Marine Science*, 60: 446–452.
- Brierley, A. S., and Cox, M. J. 2010. Shapes of krill swarms and fish schools emerge as aggregation members avoid predators and access oxygen. *Current Biology*, 20: 1758–1762.
- Brierley, A. S., and Cox, M. J. 2015. Fewer but not smaller schools in declining fish and krill populations. *Current Biology*, 25: 75–79.
- Brierley, A. S., Ward, P., Watkins, J. L., and Goss, C. 1998. Acoustic discrimination of Southern Ocean zooplankton. *Deep Sea Research Part II: Topical Studies in Oceanography*, 45: 1155–1173.

- Christin, S., Hervet, É., and Lecomte, N. 2019. Applications for deep learning in ecology. *Methods in Ecology and Evolution*, 10: 1632–1644.
- Coetzee, J. 2000. Use of a shoal analysis and patch estimation system (SHAPES) to characterise sardine schools. *Aquatic Living Resources*, 13: 1–10.
- Cohen, J. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20: 37–46.
- Cox, M. J., Watkins, J. L., Reid, K., and Brierley, A. S. 2011. Spatial and temporal variability in the structure of aggregations of Antarctic krill (*Euphausia superba*) around South Georgia, 1997–1999. *ICES Journal of Marine Science*, 68: 489–498.
- D’Elia, M., Patti, B., Bonanno, A., Fontana, I., Giacalone, G., Basilone, G., and Fernandes, P. G. 2014. Analysis of backscatter properties and application of classification procedures for the identification of small pelagic fish species in the Central Mediterranean. *Fisheries Research*, 149: 33–42.
- De Robertis, A., Lawrence-Slavas, N., Jenkins, R., Wangen, I., Mordy, C. W., Meinig, C., Levine, M., *et al.* 2019. Long-term measurements of fish backscatter from saildrone unmanned surface vehicles and comparison with observations from a noise-reduced research vessel. *ICES Journal of Marine Science*, 76: 2459–2470.
- Demer, D. A., Andersen, L. N., Basset, C., Berger, L., Chu, D., Condiotty, J., Cutter, G. R., *et al.* 2017. 2016 USA–Norway EK80 Workshop Report: Evaluation of a Wideband Echosounder for Fisheries and Marine Ecosystem Science. ICES Cooperative Research Report No. 336. 69 pp.
- Demer, D. A., Berger, L., Bernasconi, M., Bethke, E., Boswell, K. M., Chu, D., Domokos, R., *et al.* 2015. Calibration of Acoustic Instruments. ICES Cooperative Research Report No. 326. 133 pp.
- Diner, N. 2001. Correction on school geometry and density: approach based on acoustic image simulation. *Aquatic Living Resources*, 14: 211–222.
- Escobar-Flores, P. C., Ladroit, Y., and O’Driscoll, R. L. 2019. Acoustic assessment of the micronekton community on the Chatham Rise, New Zealand, using a semi-automated approach. *Frontiers in Marine Science*, 6:507.
- Escobar-Flores, P. C., O’Driscoll, R. L., and Montgomery, J. C. 2018. Predicting distribution and relative abundance of mid-trophic level organisms using oceanographic parameters and acoustic backscatter. *Marine Ecology Progress Series*, 592: 37–56.
- Fallon, N. G., Fielding, S., and Fernandes, P. G. 2016. Classification of Southern Ocean krill and icefish echoes using random forests. *ICES Journal of Marine Science*, 73: 1998–2008.
- Fernandes, P. G., Gerlotto, F., D. V. Nakken, O., and Simmonds, E. J. 2002. Acoustic applications in fisheries science: the ICES contribution. *ICES Marine Science Symposia*, 215: 483–492.
- Fernandes, P. G. 2009. Classification trees for species identification of fish-school echotraces. *ICES Journal of Marine Science*, 66: 1073–1080.
- Fleischman, S., and Burwen, D. L. 2003. Mixture models for the species apportionment of hydroacoustic data, with echo-envelope length as the discriminatory variable. *ICES Journal of Marine Science*, 60: 592–598.
- Footo, K. G., Knudsen, H. P., and Vestnes, G. 1983. Standard calibration of echo sounders and integrators with optimal copper spheres. *Fiskeridirektoratet, Havforskningsinstituttet*, 17: 335–346.
- Francis, J. M. 1996. Nature Conservation and the Precautionary Principle. *Environmental Values*, 5: 257–264.
- Gerlotto, F., Castillo, J., Saavedra, A., Barbieri, M. A., Espejo, M., and Cotel, P. 2004. Three-dimensional structure and avoidance behaviour of anchovy and common sardine schools in central southern Chile. *ICES Journal of Marine Science*, 61: 1120–1126.
- Gerlotto, F., Soria, M., and Fréon, P. 1999. From two dimensions to three: the use of multibeam sonar for a new approach in fisheries acoustics. *Canadian Journal of Fisheries and Aquatic Sciences*, 56: 6–12.
- Getabu, A., Tumwebaze, R., and MacIennan, D. N. 2003. Spatial distribution and temporal changes in the fish populations of Lake Victoria. *Aquatic Living Resources*, 16: 159–165.
- Goudswaard, P. C., Witte, F., and Katunzi, E. F. B. 2008. The invasion of an introduced predator, Nile perch (*Lates niloticus*, L.) in Lake Victoria (East Africa): chronology and causes. *Environmental Biology of Fishes*, 81: 127–139.
- Greene, C., Meyer-Gutbrod, E., McGarry, L., Hufnagle, L., Chu, D., McClatchie, S., Packer, A., *et al.* 2014. A wave glider approach to fisheries acoustics: transforming how we monitor the Nation’s Commercial Fisheries in the 21st century. *Oceanography*, 27: 168–174.
- Haralabous, J., and Georgakarakos, S. 1996. Artificial neural networks as a tool for species identification of fish schools. *ICES Journal of Marine Science*, 53: 173–180.
- Hintze, J. L., and Nelson, R. D. 1998. Violin plots: a box plot-density trace synergism. *The American Statistician*, 52: 181.
- Kayanda, R., Everson, I., Munyaho, T., and Mgaya, Y. 2012. Target strength measurements of Nile perch (*Lates niloticus*: Linnaeus, 1758) in Lake Victoria, East Africa. *Fisheries Research*, 113: 76–83.
- Klevjer, T., Tarling, G., and Fielding, S. 2010. Swarm characteristics of Antarctic krill *Euphausia superba* relative to the proximity of land during summer in the Scotia Sea. *Marine Ecology Progress Series*, 409: 157–170.
- Kolding, J., van Zwieten, P., Marttin, F., Funge-Smith, S., and Poulain, F. 2019. Freshwater Small Pelagic Fish and Fisheries in Major African Lakes and Reservoirs in Relation to Food Security and Nutrition. FAO Fisheries and Aquaculture Technical Paper No. 642. Rome. 124 pp.
- Korneliusson, R. J., Heggelund, Y., Macaulay, G. J., Patel, D., Johnsen, E., and Eliassen, I. K. 2016. Acoustic identification of marine species using a feature library. *Methods in Oceanography*, 17: 187–205.
- Kuhn, M. 2019. caret: Classification and Regression Training. R package version 6.0-84.
- Lawson, G. 2001. Species identification of pelagic fish schools on the South African continental shelf using acoustic descriptors and ancillary information. *ICES Journal of Marine Science*, 58: 275–287.
- LVFO. 2006. Lake Victoria Acoustic Survey August 2005: Report of an Analysis Workshop 3–14 October 2005 Mwanza, Tanzania and further analysis during March 2006. Jinja, Uganda. 58 pp.
- LVFO. 2012. The Standard Operating Procedures for Acoustic Surveys. Jinja, Uganda. 63 pp.
- LVFO. 2015. A Report of the Lake-Wide Hydro-Acoustic Survey 2015. Jinja, Uganda. 30 pp.
- LVFO. 2016a. Regional Catch Assessment Survey Synthesis Report June 2005 to December 2015. Jinja, Uganda. 35 pp.
- LVFO. 2016b. Regional Status Report on Lake Victoria Biennial Frame Surveys between 2000 and 2016. Jinja, Uganda. 87 pp.
- LVFO. 2018. Fisheries Management Plan III (FMP III) for Lake Victoria Fisheries 2016–2020. Jinja, Uganda. 52 pp.
- Madureira, L. S. P., Everson, I., and Murphy, E. J. 1993. Interpretation of acoustic data at two frequencies to discriminate between Antarctic krill (*Euphausia superba* Dana) and other scatterers. *Journal of Plankton Research*, 15: 787–802.
- Malde, K., Handegard, N. O., Eikvil, L., and Salberg, A. 2019. Machine intelligence and the data-driven future of marine science. *ICES Journal of Marine Science*, 77: 1274–1285.
- Mangeni-Sande, R., Taabu-Munyaho, A., Ogutu-Ohwayo, R., Nkalubo, W., Natugonza, V., Nakiyende, H., Nyamweya, C. S., *et al.* 2019. Spatial and temporal differences in life history parameters of *Rastrineobola argentea* (Pellegrin, 1904) in the Lake Victoria basin in relation to fishing intensity. *Fisheries Management and Ecology*, 26: 406–412.


- Mordy, C., Cokelet, E., De Robertis, A., Jenkins, R., Kuhn, C., Lawrence-Slavas, N., Berchok, C., *et al.* 2017. Advances in ecosystem research: saildrone surveys of oceanography, fish, and marine mammals in the Bering Sea. *Oceanography*, 30: 113–115.
- Nero, R. W., and Magnuson, J. J. 1989. Characterization of patches along transects using high-resolution 70-kHz integrated acoustic data. *Canadian Journal of Fisheries and Aquatic Sciences*, 46: 2056–2064.
- Njiru, M., Nyamweya, C., Gichuki, J., Mugidde, R., Mkumbo, O., and Witte, F. 2012. Increase in anoxia in Lake Victoria and its effects on the fishery. *In Anoxia*, pp. 99–128. Ed. by P. Padilla. InTech, Rijeka.
- Odongkara, K., Yongo, E., and Mhagama, F. 2016. The State of Lake Victoria Dagua *Rastrineobola argentea*: Quantity, Quality, Value Addition, Utilization and Trade in the East African Region, for Improved Nutrition, Food Security and Income. Report of the EAC and Lake Victoria Fisheries Organisation. 87 pp.
- Okedi, J. 1981. The Engraulicrpris “dagaa” fishery of Lake Victoria with special reference to the southern waters of the lake. *In Proceedings of the Workshop of the Kenya Marine and Fisheries Research Institute on Aquatic Resources of Kenya*, 13–19 July. Mombasa.
- Paramo, J., Gerlotto, F., and Oyarzun, C. 2010. Three dimensional structure and morphology of pelagic fish schools. *Journal of Applied Ichthyology*, 26: 853–860.
- Petitgas, P. 2001. Geostatistics in fisheries survey design and stock assessment: models, variances and applications. *Fish and Fisheries*, 2: 231–249.
- Proud, R., Cox, M. J., and Brierley, A. S. 2017. Biogeography of the global ocean’s mesopelagic zone. *Current Biology*, 27: 113–119.
- Proud, R., Cox, M. J., Wotherspoon, S., and Brierley, A. S. 2015. A method for identifying sound scattering layers and extracting key characteristics. *Methods in Ecology and Evolution*, 6: 1190–1198.
- Reid, D., Scalabrin, C., Petitgas, P., Masse, J., Aukland, R., Carrera, P., and Georgakarakos, S. 2000. Standard protocols for the analysis of school based data from echo sounder surveys. *Fisheries Research*, 47: 125–136.
- Ripley, B. 2019. tree: Classification and Regression Trees. R package version 1.0-40.
- Scalabrin, C., Diner, N., Weill, A., Hillion, A., and Mouchot, M. C. 1996. Narrowband acoustic identification of monospecific fish shoals. *ICES Journal of Marine Science*, 53: 181–188.
- Sharpe, D. M. T., and Chapman, L. J. 2014. Niche expansion in a resilient endemic species following introduction of a novel top predator. *Freshwater Biology*, 59: 2539–2554.
- Simmonds, E. J., and MacLennan, D. N. 2005. *Fisheries Acoustics: Theory and Practice*. 2nd edn. Blackwell Science, Oxford. 437 pp.
- Simmonds, E. J., Armstrong, F., and Copland, P. J. 1996. Species identification using wideband backscatter with neural network and discriminant analysis. *ICES Journal of Marine Science*, 53: 189–195.
- Stone, M. 1974. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society: Series B (Methodological)*, 36: 111–133.
- Strobl, C., Boulesteix, A.-L., Kneib, T., Augustin, T., and Zeileis, A. 2008. Conditional variable importance for random forests. *BMC Bioinformatics*, 9: 307.
- Strobl, C., Hothorn, T., and Zeileis, A. 2009. Party on! *The R Journal*, 1: 14.
- Taabu-Munyaho, A., Marshall, B. E., Tomasson, T., and Marteinsdottir, G. 2016. Nile perch and the transformation of Lake Victoria. *African Journal of Aquatic Science*, 41: 127–142.
- Taabu-Munyaho, A., Nyamweya, C. S., Sitoki, L., Kayanda, R., Everson, I., and Marteinsdóttir, G. 2014. Spatial and temporal variation in the distribution and density of pelagic fish species in Lake Victoria, East Africa. *Aquatic Ecosystem Health & Management*, 17: 52–61.
- Tarling, G. A., David, P., Guerin, O., and Buchholz, F. 2001. The Swarm Dynamics of Northern Krill (*Meganyctiphanes norvegica*) and Pteropods (*Cavolinia inyexa*) during Vertical Migration in the Ligurian Sea Observed by an Acoustic Doppler Current Profiler, 16: 1–16.
- Tegowski, J., Gorska, N., and Klusek, Z. 2003. Statistical analysis of acoustic echoes from underwater meadows in the eutrophic Puck Bay (southern Baltic Sea). *Aquatic Living Resources*, 16: 215–221.
- Tumwebaze, R. 2003. Hydroacoustic Abundance Estimation and Population Characteristics of *Rastrineobola argentea* in Lake Victoria. University of Hull, Hull, UK.
- Wanink, J. H. 1999. Prospects for the fishery on the small pelagic *Rastrineobola argentea* in Lake Victoria. *Hydrobiologia*, 407: 183–189.

Handling editor: Olav Godo

Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Original Article

Acoustic classification in multifrequency echosounder data using deep convolutional neural networks

Olav Brautaset¹, Anders Ueland Waldeland¹, Espen Johnsen², Ketil Malde², Line Eikvil¹, Arnt-Børre Salberg¹, and Nils Olav Handegard  ^{2*}

¹Norwegian Computing Center, P.O. Box 114 Blindern, Oslo 0314, Norway

²Institute of Marine Research, Nordnesgaten 50, Bergen 5005, Norway

*Corresponding author: tel: + 47 95854057; e-mail: nilsolav@hi.no.

Brautaset, O., Waldeland, A. U., Johnsen, E., Malde, K., Eikvil, L., Salberg, A.-B., and Handegard, N. O. Acoustic classification in multifrequency echosounder data using deep convolutional neural networks. – ICES Journal of Marine Science, 77: 1391–1400.

Received 5 July 2019; revised 5 November 2019; accepted 14 November 2019; advance access publication 21 January 2020.

Acoustic target classification is the process of assigning observed acoustic backscattering intensity to an acoustic category. A deep learning strategy for acoustic target classification using a convolutional network is developed, consisting of an encoder and a decoder, which allow the network to use pixel information and more abstract features. The network can learn features directly from data, and the learned feature space may include both frequency response and school morphology. We tested the method on multifrequency data collected between 2007 and 2018 during the Norwegian sandeel survey. The network was able to distinguish between sandeel schools, schools of other species, and background pixels (including seabed) in new survey data with an F1 score of 0.87 when tested against manually labelled schools. The network separated schools of sandeel and schools of other species with an F1 score of 0.94. A traditional school classification algorithm obtained substantially lower F1 scores (0.77 and 0.82) when tested against the manually labelled schools. To train the network, it was necessary to develop sampling and preprocessing strategies to account for unbalanced classes, inaccurate annotations, and biases in the training data. This is a step towards a method to be applied across a range of acoustic trawl surveys.

Keywords: acoustic classification, big data, deep learning, machine learning, sandeel

Introduction

Acoustic trawl surveys (Simmonds and MacLennan, 2005) are commonly used in fisheries assessments to provide data that support advice on total allowable catches for a wide range of fish stocks. Echosounders are instruments that produce soundwaves and record the intensity of backscattered soundwaves produced by targets in the water column. Echosounder observations are calibrated (Foote *et al.*, 1987) and can be integrated over specific depth ranges to calculate the area-backscattering coefficient (the average backscattering intensity per metre square; MacLennan *et al.*, 2002). The area-backscattering coefficient is linearly related to fish abundance (Foote, 1983) for a given representative target strength (Ona, 2003), and under the assumption that the backscattering intensity can be correctly assigned to a species or a

group of species. The categorization of species into groups is aided by trawl samples of the fish, which are used to estimate the age and length distributions of fish populations. This method is typically used for pelagic or semi-pelagic species, such as walleye pollock (Karp and Walters, 1994), herring, blue whiting (Gastauer *et al.*, 2016), capelin (Gjosæter *et al.*, 2015), and sandeel (Johnsen *et al.*, 2009).

The process of assigning values of acoustic backscattering intensity to an acoustic category or group is typically a manual operation. An operator, based on information discerned from trawl catches, multifrequency echosounder observations, and any other auxiliary information, assigns values of acoustic backscattering intensity to an acoustic category, which can represent a species or a group of species. The process is typically time-

consuming and often incurs operator-based biases (Simmonds and MacLennan, 2005). To reduce bias and increase efficiency, several features ascertained from the acoustic observations have been used to aid, automate, or partially automate the process (Korneliussen, 2018). In addition to trawl sampling, features such as the location and position, environmental variables, and acoustically derived morphometric and energy features may also have discriminatory power (e.g. Horne, 2000; Reid, 2000). The main feature used in species classification is the relative frequency response, i.e. the fraction of backscattering intensity observed at one frequency relative to a reference frequency, typically 38 kHz (Kloser et al., 2002; Korneliussen and Ona, 2003). Based on these features, different methods have been used to classify values of backscattering intensity, including Bayesian methods (Korneliussen et al., 2016), semi-supervised methods (Woillez et al., 2012), and machine learning methods including random forest (Proud et al., 2020; Fallon et al., 2016) and artificial neural networks (Haralabous and Georgakarakos, 1996).

The current methods require that the feature space used for the classification is predefined, e.g. averaging the relative frequency response over a suitable number of pixels or defining the most efficient morphometric features, but this step is not trivial, i.e. how much should we smooth and what are the best morphological shapes? Defining the feature space for broadband fisheries echosounders (Mukai and Amakasu, 2016), where small-scale features in the frequency response may have large discriminatory power, may be even more challenging. A method that combines the feature extraction with the classification is preferable.

In recent years, deep convolutional neural networks (CNNs) have emerged as the leading modelling tools for image classification, segmentation, and semantic mapping both generally (Hariharan et al., 2015; Long et al., 2015) and also within marine science (Malde et al., 2020). CNNs do not require features to be designed in advance as they can learn the appropriate features from “raw” data, like images, and they have been shown to be superior in solving problems in computer vision and image analysis (Russakovsky et al., 2015). A CNN consists of a sequence of operations, referred to as *layers*, applied to the input image. The output from one layer is thus the input to the subsequent layer. Each layer typically consists of a number of separate convolutions with small *filter kernels*, followed by some non-linear function, and may also be combined with other operations. Each filter kernel consists of a number of coefficients, and using gradient-based optimization, these filter coefficients are tuned to minimize the classification error on annotated training data, referred to as *training* (Rumelhart et al., 1986). During training, the first layers will typically learn to recognize edges, lines, and corners, and the later layers can represent more abstract features. With this approach, the network can use the raw data directly as opposed to the traditional approach where the features must be predefined. Training a CNN requires large amounts of training data, i.e. ground truth data with corresponding annotations.

Image segmentation using CNNs can be carried out using several different approaches. One strategy is to train a classifier on small image patches and then either classify all pixels using a sliding window approach, or more efficiently, by converting the fully connected layers in the CNN to convolutional layers (Sermanet et al., 2013), thereby avoiding overlapping computations. Another approach is pixel-to-pixel semantic mapping using end-to-end learning (Long et al., 2015). It uses a fully convolutional network (FCN), consisting of an encoder and a decoder, where

the encoder maps the image to a low-resolution representation and the decoder provides a mapping from the low-resolution representation to the pixel-wise representation. An FCN has the advantage that the input size can vary since the convolutions “slide” over the data set, as opposed to networks that have fully connected layers requiring a fixed input size. A popular network architecture for semantic mapping is the U-Net (Ronneberger et al., 2015), characterized by skip connections between the corresponding encoder and decoder layers.

The objectives of this article are to (i) develop a deep learning strategy that is suitable for segmenting and classifying echosounder data collected during acoustic trawl surveys without prior feature extraction; (ii) demonstrate that the strategy developed in (i) works on a real test case, and (iii) provide perspectives, e.g. pros and cons, on the use of deep learning algorithms in the classification of acoustic observations into acoustic categories (e.g. species groups).

Material and methods

The sandeel survey

Data collected during the Norwegian North Sea Sandeel survey were used as test case for this study (ICES, 2016). The lesser sandeel (*Ammodytes marinus*), hereafter sandeel, is a small fish that does not have a swim bladder. For large parts of its life the sandeel hides by burrowing in sandy seabed, where the proportion of fine silt and clay particles is low (Macer, 1966; Wright et al., 2000). During the feeding season in spring, adults that have burrowed into the sandy substrate at night emerge at dawn (Winslade, 1974) to form large schools in the upper pelagic zone and predate on zooplankton (Freeman et al., 2004; Johnsen et al., 2017). The sandeel is a key species in the North Sea ecosystem, being a major prey species for several predators, including sea birds, seals, and larger fish (Furness, 2002), and is also a valuable target for commercial fishing.

The Institute of Marine Research, Norway, has been conducting acoustic trawl surveys for sandeel during April and May since 2005 in the sandeel areas of the north eastern part of the North Sea (Johnsen et al., 2017). The survey series (2005–2018) was conducted using the RV Johan Hjorth (2005–2008, 2010–2011), RV GO Sars (2009), FV Brennholm (2012), and FV Eros (2013–2018). All vessels were equipped with multifrequency Simrad EK60 echosounder systems operating transducers at 18, 38, 120, and 200 kHz, except for the FV Brennholm (2012) that was without a 120-kHz EK60 echosounder but collected 120 kHz using a Simrad ME70 sonar (Trenkel et al., 2008). In addition, the RV GO Sars and FV Eros (from 2014) were equipped with a 70- and 333-kHz echosounder. The echosounders were calibrated in accordance with standard procedures before each survey (Foote et al., 1987). During operation, the pulse duration and ping repetition frequency were set to 1.024 ms and 3–4 Hz, respectively, for all frequencies and vessel speed was kept at approximately ten knots. Echosounder observations were stored as values of volume backscattering coefficient (s_v ; average amount of backscattering intensity per cubic metre) by frequency (MacLennan et al., 2002). See Johnsen et al., 2009 for further details.

Data preprocessing

In some instances, the pulse duration (i.e. range resolution) and ping rate differed from the standard settings. To be consistent, the data were interpolated into a common time-range grid based

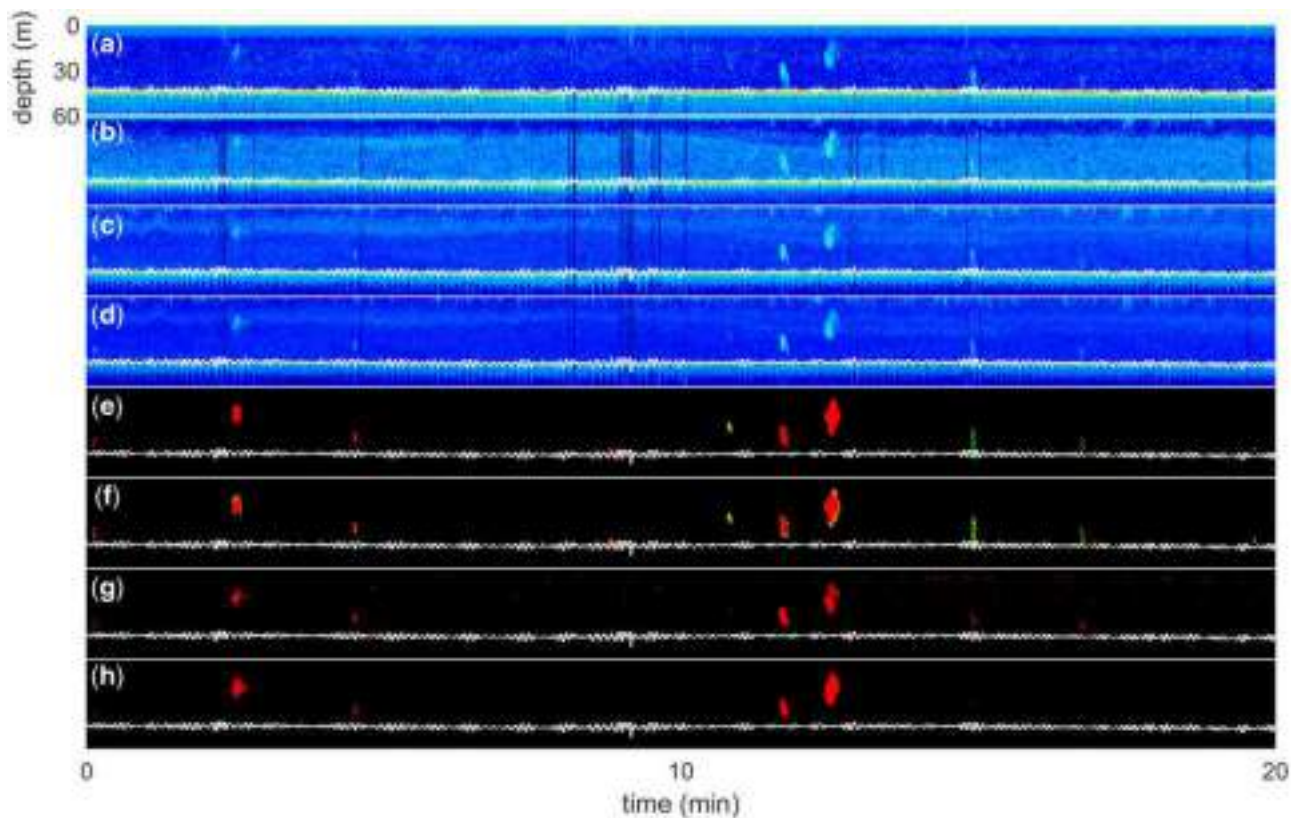


Figure 1. Echogram with four frequency channels (a–d, 18, 38, 120 and 200 kHz) and original (e) and modified (f) annotations, where black is the “background” class, red (grey in print) is the “sandeel” class, green (dark grey in print) is the “other” class, and yellow (light grey in print) is the “ignore” pseudo class, the predictions from the benchmark method (g), and the predictions from our method (h). Here, black and red (grey in print) are the background/other and sandeel classes, respectively. The seabed is shown as a white curve in all panels.

on the resolution of the 200-kHz data. The median ping rate was used to detect missing pings, and, when missing pings occurred, columns of zeros (mapped to -75 dB re 1 m^{-1} after log transformation) were inserted into the s_v data. If the range vector of the other frequencies was of a lower resolution, the data were interpolated onto the 200-kHz range vector. If the range vector had a higher resolution, the s_v values were averaged into bins defined by the 200-kHz range vector. This resulted in s_v values in a uniform time-range grid (Figure 1a–d), similar to pixels in a four-channel image, and we refer to these values as pixels hereafter. The seabed was approximately located as the depth with maximum increase in vertical gradient for each ping. This was used for balanced sampling (see below) and to avoid false predictions.

The survey series uses “sandeel”, “other”, “0-group sandeel”, and “possible sandeel” as acoustic categories, denoted “classes” hereafter, that were manually annotated by the same operator across all years. The annotations were interpolated into a pixel map corresponding to the echosounder data, and each pixel was allocated to one class. The acoustic classes “other” and “sandeel” have been used for all years, and the “sandeel” class is the only class used in official survey estimates. In addition, “possible sandeel” was introduced for schools where the frequency response was not consistent with sandeel but where the operator was in doubt and, for the 2016 survey, the “0-group sandeel” was introduced due to an extraordinary high density of juveniles. Each school varied from a few metres in length and height to >1 km in length extending across large parts of the water column (Johnsen

et al., 2017). The 200-kHz data were used as the primary frequency during annotation since it has the highest sandeel signal-to-noise ratio, and each school was annotated and classified by acoustic category using the Large Scale Survey System (LSS) postprocessing software (Korneliusson *et al.*, 2016). The manual annotations were mainly based on the frequency response of each school (see Johnsen *et al.*, 2009) and validated by trawl samples where applicable. The “0-group sandeel” and “possible sandeel” classes were added to an “ignore” pseudo class, and all other pixels not associated with a class were set to “background”. This resulted in pixel-based annotations with classes “sandeel”, “other”, “background”, and “ignore” (Figure 1e). Note that the bottom echo is included in the “background” class. Table 1 shows the total number of schools for each class.

The purpose of the annotations is to estimate sandeel abundance, which is calculated by summing up the 200-kHz backscattering intensity (Figures 1d and 2a) of the sandeels over a given region and dividing by their mean target strength. Heave measurements of the survey vessel were used to correct the echogram data and annotations. However, all figures are presented without heave corrections. The annotations were often coded as rectangular bounding boxes (when viewed with heave corrections; Figure 2b), and a portion of the bounding box would, consequently, include background pixels. This does not substantially affect the abundance estimate since adding low-value pixels does not substantially contribute to the total integrated backscattering intensity, but it may confuse a pixel-based classifier trying to

Table 1. Number of schools annotated as “sandeel”, “other”, and “ignore” per year in the final dataset.

Year	Sandeel schools	Other schools	Ignored schools
2007	453	605	0
2008	1 664	4 378	0
2009	699	2 755	30
2010	3 206	2 560	542
2011	623	1 685	177
2013	2 015	5 133	527
2014	1 121	6 113	549
2015	1 515	4 866	523
2016	829	4 423	2 130
2017	3 602	2 362	755
2018	4 678	1 917	255
Total	20 405	36 797	5 488

Table 2. Sampling strategy for drawing random $4 \times 256 \times 256$ crops for training.

Classes	Probability	Description
Background	1/26	Random crop from area without fish, above the seabed
Seabed	5/26	Random crop from area containing seabed
Sandeel	5/26	Random crop from area containing “sandeel” class
Other	5/26	Random crop from area containing “other” class
Seabed + sandeel	5/26	Random crop from area containing both seabed and “sandeel” classes
Seabed + other	5/26	Random crop from area containing both seabed and “other” classes

We divided regions of the echograms into these six classes and drew random samples from each class with the given probabilities.

predict the “background” class. To amend this problem, we modified the original annotations based on the s_v values. Any pixel annotated as “sandeel” or “other” with a corresponding 200-kHz s_v value outside the interval $[10^{-7} \text{ m}^{-1}, 10^{-4} \text{ m}^{-1}]$ was assigned to the “ignore” pseudo class (Figure 2). We set the threshold values based on a visual inspection of multiple echograms. We further smoothed the fish annotated pixel regions by applying binary morphological closing to the modified “sandeel” and “other” annotations, using a 7×7 disk-shaped structure element (Figure 2c).

Training data

For each survey, the acoustic data were comprised of a single continuous echogram for each frequency, but for training purposes, we divided each dataset into $4 \times 256 \times 256$ pixels crops, where 4 is the number of frequencies. We also applied a decibel transform to the s_v values and applied a hard threshold to values below $-75 \text{ dB re } 1 \text{ m}^{-1}$ and above $0 \text{ dB re } 1 \text{ m}^{-1}$. Each annotated echogram also possessed a heavy class imbalance; there were many more “background” pixels (99.8%) than “sandeel” (0.1%) and “other” pixels (0.1%). To expose the network to enough samples with fish schools when training, we first created an algorithm to get crops that were composed entirely of background pixels and similarly for crops that included “sandeel” and “other” pixels, respectively. We then balanced the dataset by applying an

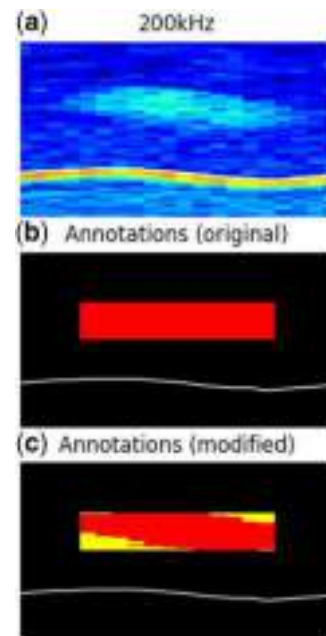


Figure 2. (a) Small patch from an echogram (200-kHz channel) with (b) original and (c) modified annotations. Modified annotations were obtained from original annotations using thresholds on the 200-kHz channel followed by morphological closing. The classes “background”, “sandeel”, and “ignore” are presented in black, red, and yellow (black, grey, and light grey in print), respectively. Axes are similar to Figure 1, where the vertical and horizontal axes represent depth and time, respectively.

equal sampling probability to crops containing seabed only, “sandeel”, “other”, seabed and “sandeel”, and seabed and “other” (see Table 2). All these crop types include the “background” class, but, in addition, we randomly sampled a smaller fraction of crops that had “background” pixels only (see Table 2). In addition, most of the sandeel schools resided close to the seabed and the balanced sampling during training mitigated the network from classifying all schools close to the seabed as sandeel, or worse, classifying the bottom itself as sandeel.

We partitioned the dataset into a training and validation dataset and a test dataset by different years, where 2011–2016 was used for training and validation and 2007–2010 combined with 2017–2018 was used for testing. From the training and validation set, we used 85% randomly drawn echograms for training and the remaining 15% for validation to select the best model. Among the test sets, the final year (2018) was unseen until the final evaluation.

Deep learning model and training

In this study, we built a classifier that was based on a slightly modified version of the U-Net architecture (Ronneberger et al., 2015). The U-Net is a pixel-wise image segmentation network with a convolutional encoder–decoder architecture (Figure 3 and Supplementary Tables S1 and S2), originally developed for the segmentation of medical images. An encoder–decoder architecture can represent both pixel-wise and abstract features simultaneously. Our modified U-Net takes four frequency channels, 18, 38, 120, and 200 kHz, and a 256×256 range-time subset of the echogram as the input ($4 \times 256 \times 256$), and “encode” it to a

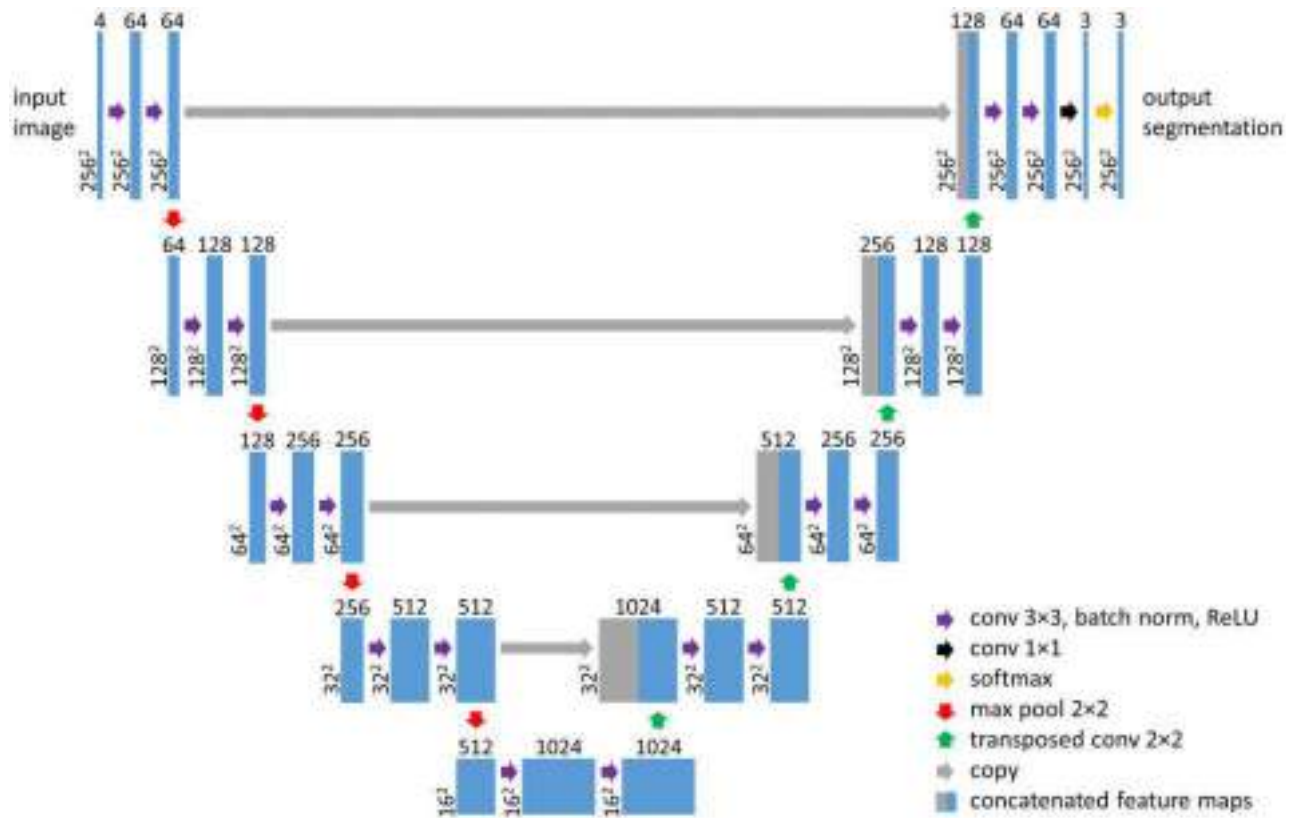


Figure 3. The network architecture, a slightly modified version of the original U-Net. The input is the $4 \times 256 \times 256$ crops, and the $3 \times 256 \times 256$ output is the softmax for each pixel by class (“sandeel”, “other”, and “background”).

16×16 “image” with 1024 abstract features ($16 \times 16 \times 1024$). The decoder then takes these features and generates (decode) an output for the classes “background”, “other”, and “sandeel” ($3 \times 256 \times 256$) for each of the input pixels. The architecture also copies the lower level features at each step when decoding, resulting in the decoder to both have access to low-level features (e.g. the frequency response in a small region) and more abstract features (e.g. like the overall shape). Finally, the output is passed through a “softmax” function where each of the three output classes is mapped to the interval $[0, 1]$ and add up to 1, like a probability for each class for each pixel. Contrary to the original implementation, we inserted a batch normalization layer (Ioffe and Szegedy, 2015) between each convolutional layer and its subsequent activation function to reduce covariate shift, i.e. normalizing the distribution of outputs from each convolutional layer.

We trained the model over 5000 iterations using batches of 16 random $4 \times 256 \times 256$ crops. We used random uniform weight initialization and optimized with stochastic gradient descent with initial learning rate 0.01 and momentum 0.95. The learning rate controls how much the model parameters can change in each training iteration, while the momentum controls how much a training sample will influence the change of model parameters in the subsequent iterations. The learning rate was reduced by a factor of 0.5 every 1000 iterations. The model was evaluated on the validation set every 20th iteration. Due to the class imbalance, we used a weighted cross entropy loss with class weights (background = 1, sandeel = 30, and other = 25) to further adjust for imbalanced classes, giving less weight to each background pixel to

compensate for this class being more frequently observed. We randomly flipped the training crops about the vertical axis and added random multiplicative noise to a random 5% of the pixels. The hyperparameters were set by training the model multiple times, each time with a different combination of hyperparameters. We observed the impact on classification accuracies on the training and validation set for different combinations and fine-tuned the hyperparameters further based on the combination that gave the best initial results.

Since the network is based on convolutions only, the input image can be of any size during prediction and does not have to resemble the $4 \times 256 \times 256$ crops used for training. When using the network for prediction, we applied it to tiled segments (corresponding to the echosounder raw files) of the full survey echograms, including an overlap between segments of 40 pixels to avoid edge effects. As a postprocessing step, we also removed any predictions of fish more than ten pixels below the seabed.

Due to the heavy class imbalance, we used precision/recall curves rather than receiver operating characteristic curves to evaluate the performance. The network is considered a binary pixel classifier (positive/negative) by fixing a *threshold* value between 0 and 1, classifying a pixel as positive if the network output for the “sandeel” class is above this threshold value and negative otherwise. Using “sandeel” as the positive class, the precision is the proportion of *predicted* “sandeel” pixels that are correct, and the recall is the fraction of “sandeel” labels that are correctly predicted as “sandeel”. By predicting all pixels as “sandeel”, recall would be

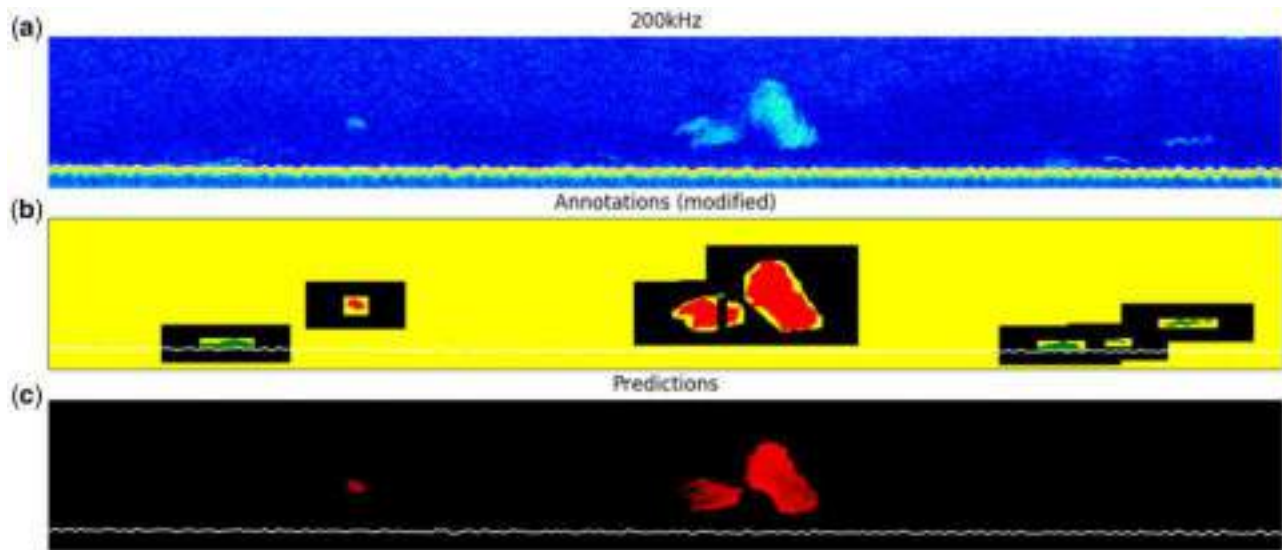


Figure 4. Illustration of evaluated pixels for computing precision/recall curves. (a) The 200-kHz echogram, (b) modified annotations where yellow pixels (light grey in print) are the ignore pseudo class, while in this example, “sandeel” (red, grey in print) is treated as positive and “other” (green, dark grey in print) and “background” (black) are regarded as negatives when calculating the precision/recall curves. (c) The predictions of the “sandeel” class where a high softmax is shown as bright red and a low softmax is shown as dark red (grayscale in print). Axes are similar to Figure 1, where the vertical and horizontal axes represent depth and time, respectively.

1, but precision would be low, and conversely, by correctly predicting one pixel as sandeel, precision would be 1, but recall would be low. Varying the threshold value results in different precision and recall values, where the recall may increase at the cost of lowering the precision. For a good classification, both precision and recall should be high and the F1 score at a given threshold value, defined as

$$F1 = \left(\frac{\text{precision}^{-1} + \text{recall}^{-1}}{2} \right)^{-1},$$

is typically used to test a methods performance. In our case, we only report the maximized F1 score, i.e. choosing the threshold value that gives the highest F1 score.

When evaluating the performance, we used two slightly different approaches when calculating the precision/recall curves for the background class. The first approach classifies echograms using all the echogram pixels, whereas the other approach evaluated echogram regions that were within 20 pixels of any original school annotation (c.f. Figure 4). The rationale behind using these two approaches was that we suspected that a proportion of schools was not classified during annotation, and therefore, comparing “sandeel” predictions to annotations for entire echograms may result in a high number of erroneous false positives. This would again yield poor precision/recall curves and not reflect the actual performance of the model.

When calculating the precision/recall curves, we used different combinations of classes as positives and negatives, i.e. “sandeel” as positive vs. “other” as negative to test the ability to separate species given a school is detected and “sandeel” vs. “other” and “background” to test the overall ability to detect sandeel schools, which is the purpose of the survey. Predictions of the “ignore” pseudo class were not considered when calculating the curves (c.f. Figure 4).

Evaluation

To test our approach against a traditional automated processing pipeline, we used the Sandeel case in Korneliussen *et al.* (2016) as a benchmark. This was implemented as a Korona processing pipeline in LSSS and consisted of a range of operations, including noise filtering (spike noise, spot noise) smoothing, bottom detection, thresholding, school detection, and categorization. We used the exact same setup and parameters as used by Korneliussen *et al.* (2016). The categorization was exported to a file and imported and treated similarly as the predictions from the U-Net algorithm, except that the threshold for accepting a pixel as sandeel was fixed, resulting in one point in the prediction recall plot as opposed to the curves from our method. The testing was only performed in the years that we used as test cases, i.e. our network had never seen the data where we compare the methods.

Results

We trained and validated the model using echograms derived from 2011 to 2016 survey data and tested the trained model using echograms derived from 2007 to 2010 and 2017 to 2018 survey data. Figure 1h shows an example of classification based on model predictions for a four-channel echogram. In this example, the trained network successfully separated the sandeel schools from other types of fish and the background class. Figure 1g shows the corresponding classification based on the benchmark method.

The network’s ability to discriminate “sandeel” (positive) vs. “background” and “other” (negative) is good (F1 score 0.87, Figure 5) when excluding background pixels that are at a distance of 20 pixels or more from the school annotations. In this case, a total number of 170 million pixels were evaluated (positives and negatives) and the annotations consisted of 90% “background”, 6% “sandeel”, and 4% “other” (Supplementary Table S3). This resulted in an overall F1 score of 0.87 for the overall test set across

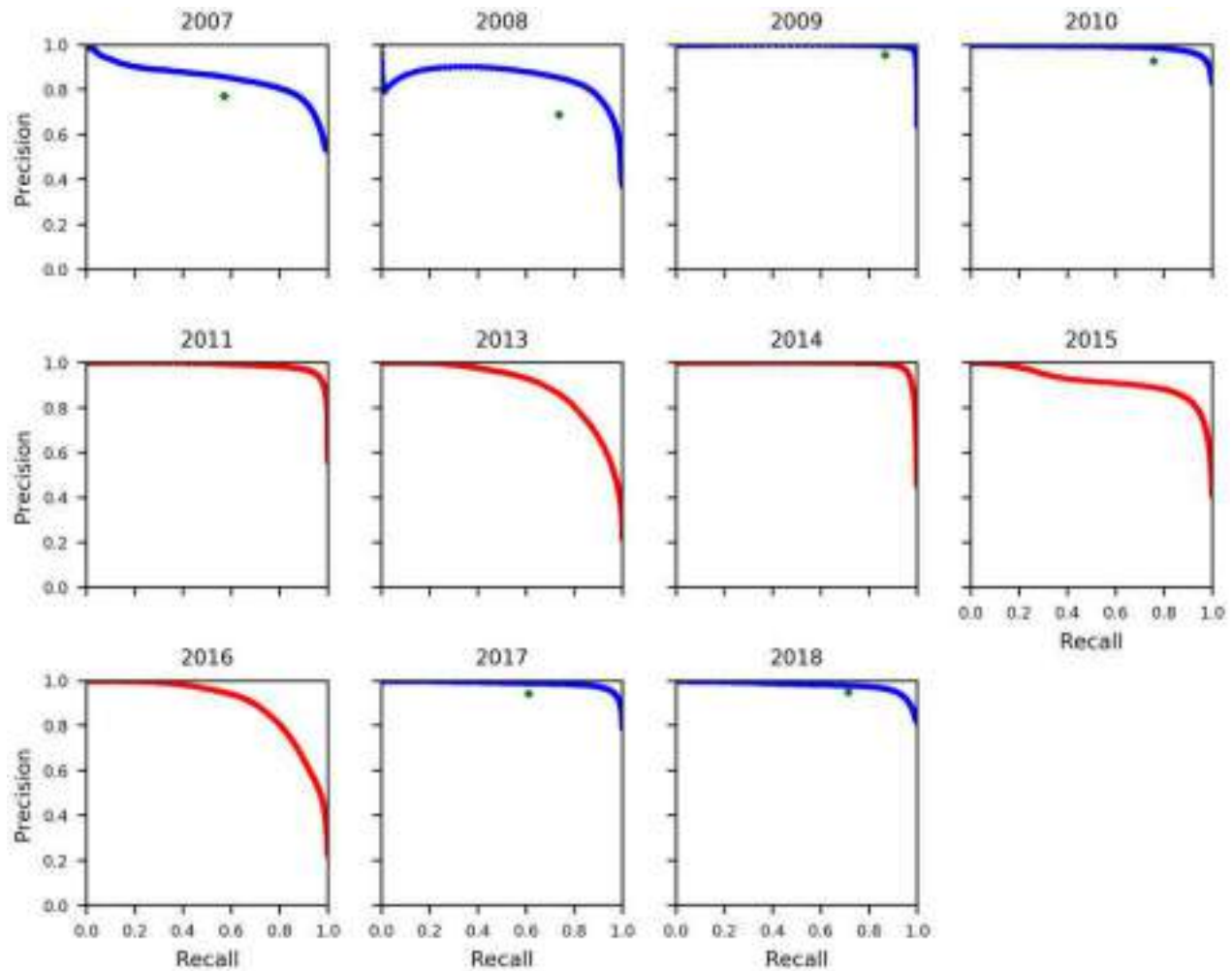


Figure 5. Precision/recall curves per year, where “sandeel” (positive) is compared to “other” and “background” (negatives) in a 20-pixel region extending beyond the original school annotations. The remaining pixels annotated as “background” or “ignore” were excluded. The red and blue curves (dark grey and light grey in print) denote the training data (years 2011–2016) and test data (years 2007–2010 and 2017–2018), respectively. Each diamond denotes the corresponding precision/recall value for the benchmark method (evaluated on test data years only).

years, with a corresponding threshold, precision, and recall of 0.80, 0.85 and 0.89, respectively. For the training and validation set, the years 2013, 2015, and 2016 did not perform as well when compared to the other years, and for the test set, the years 2007 and 2008 did not perform as well as the other years. The benchmark method achieved an overall F1 score of 0.77 for the overall test set across years, with a corresponding precision and recall of 0.80 and 0.74, respectively (Figure 5).

We also tested the network’s ability to discriminate between “sandeel” (positive) and “other” (negative) while excluding both “background” and the pseudo-class “ignore”, i.e. the ability to determine the species given that a school is detected. In this case, a total number of 18 million pixels were evaluated (positives and negatives) and the annotation consisted of 0% background, 57% “sandeel”, and 43% “other”. Our model’s separation of sandeel vs. other species obtained an overall F1 score of 0.94 for the test set. The corresponding threshold, precision, and recall were 0.50, 0.93, and 0.95, respectively. The test set results by year were also more consistent than the previous case (including background pixels), with the exception of 2007 and 2008, indicating that the network is well suited to differentiate between species

(Supplementary Figure S1). The benchmark method achieved an overall F1 score of 0.82 for the test set, with a corresponding precision and recall of 0.91 and 0.74, respectively (Supplementary Figure S1).

Our model did not perform as well when tested using entire echograms as input (Supplementary Figure S2). The performance on the test set for the years 2017 and 2018 was satisfactory (F1 score 0.61 and 0.78, respectively) but was substantially poorer for earlier years 2007–2010 (F1 score 0.11, 0.51, 0.78, and 0.68, respectively). The benchmark method achieved even lower F1 scores, both for the years 2017 and 2018 (0.32 and 0.62, respectively) and for the years 2007–2010 (0.03, 0.07, 0.42 and 0.50, respectively; Supplementary Figure S2). When looking into these specific results in more detail, we found two main reasons for the discrepancies, including missing annotations, incomplete annotations, and erroneous predictions close to the sea surface.

Missing annotations were found in several echograms, and an example of this is provided in Figure 6c, where the entire right-hand side of the echogram does not contain any annotations of fish. On closer inspection of the 200-kHz echogram (Figure 6a), clear fish marks were not annotated. In these circumstances,

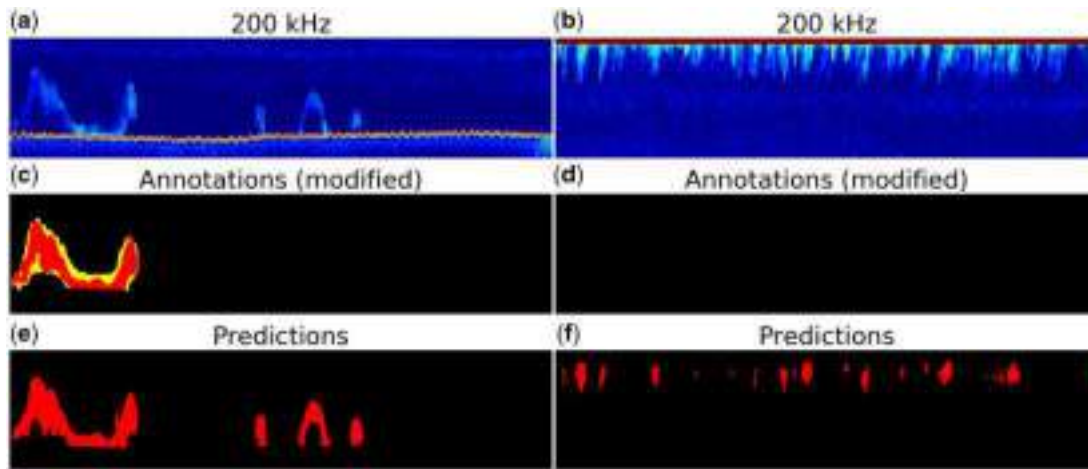


Figure 6. (a and b) The 200-kHz echograms. (c and d) Modified “sandeel” annotations in red (grey in print), the “ignore” pseudo class in yellow (light grey in print), and the “background” class in black. (e and f) Prediction of the “sandeel” class. (a, c, and e) Echogram with the absence of fish annotations in the right-hand side of the image. (b, d, and f) Echogram with false-positive predictions of sandeel close to the surface, possibly due to a zooplankton layer that the network is not trained to recognize. Axes are similar to Figure 1, where the vertical and horizontal axes represent depth and time, respectively.

positive predictions made by the model will be penalized when calculating the precision/recall curves. This illustrates a common problem encountered in the data, where image data are recorded with an abrupt absence of annotations (the remaining part of the echogram is annotated as “background”), c.f. the discussion for a possible explanation.

In some cases, the model makes false-positive predictions for “sandeel”. This is a common problem near the surface, where the model often classifies high s_v values, which could be caused by dense plankton layers, as “sandeel”. This class is annotated as “background” during training, but since we did not balance the training dataset for this case, as we did for the bottom, the model was not exposed to these “background” layers. The model has, consequently, not learned to annotate it as “background” and occasionally erroneously classifies them as “sandeel” instead.

Discussion

The objectives of this article were to define, train, and apply a deep CNN model that performs automatic classification of labelled multifrequency echosounder data and discuss how deep CNNs may be utilized for acoustic data. One of the main strengths of this model is that it does not require prior feature extraction steps, as it works directly on the output from the echosounder. These learned features may be both energetic and morphometric (Reid, 2000; Korneliussen, 2018), and there is no need to specify the features explicitly or to what degree one or the other should be used. The method also avoids any pixel averaging by school or region before applying the classifier, as the method works on high-resolution data. As with all neural networks, model interpretation is difficult. In its current design, the CNN does not provide information relating to feature importance, making it less transparent when compared to conventional methods (e.g. random forest) that work using hand-crafted features.

The manual annotations from survey data may be uncertain, and the uncertainty is not explicitly coded within the data. When using predefined features, the number of parameters in the classification model is typically lower than what is needed for CNNs. In those cases, it has been recommended to use a high-quality

training set where classifications are certain (Korneliussen *et al.*, 2016). Training a CNN requires a large amount of training data, and utilization of the full set of annotations from the survey may be needed. This has the drawback that low-quality annotation data may be used in training and validation but has the advantage that the data span the full variability across the survey. To some extent, we worked around this by adding the “possible sandeel” class to the “ignore” pseudo class. We recommend that future implementations use a combination of the above and assign a larger weight to annotations that have high certainty, e.g. those from a feature library (Korneliussen *et al.*, 2016), or allocate them to the test set only.

Using non-standard image data with annotations not made specifically for machine learning is a challenge. The annotations from the survey were designed for integrating sandeel backscattering intensity values, and assigning low s_v values to the sandeel class does not substantially contribute to the integrated sandeel backscatter. Consequently, using square bounding boxes that include background pixels does not substantially affect the integrated backscatter and is more efficient during manual annotation than drawing the school outlines. This represented a challenge in this study as the objective was to separate sandeel and background classes, and hence, refining the annotations was necessary. The modified annotations were important in making the method work. Modification of manually annotated acoustic observations may be a necessary step when using annotations to build automatic classification models such as CNNs.

Addressing the class imbalance by exposing the network to balanced mini batches of the data that contained all classes was necessary. The “other” and “sandeel” classes could be balanced, since they were annotated, but balancing the “background” class was more challenging. This class was a combination of seabed, plankton layers, empty water, and any other unknown scatterers. For the seabed, we solved this by balancing the training set with respect to crops close to seabed (since we had the bottom approximately detected), but we did not balance this for the unlabelled surface layer. This layer is most likely composed of near-surface phytoplankton blooms, specifically high densities of the gas

producing *Phaeocystis*, which produce high levels of acoustic backscattering intensity at 18 and 38 kHz. Since there is some overlap in the backscattering intensity of the surface plankton layer and of sandeel schools, the network would occasionally misclassify the “background” class as “sandeel”. A possible solution to this problem could have been to have implemented an unsupervised segmentation of the background class and then balance the training dataset based on the resulting classes. Consequently, addressing class imbalance is important for the *actual* classes in the data, not only for those that are annotated, and represents a general challenge when implementing supervised methods on acoustic data.

Processing the whole survey time series using the same automated algorithm is more efficient, consistent, and cost-effective than processing the data manually. We deliberately separated the training and test dataset by years to see if the network could generalize across years. The results showed that the performance changed by year, but this was not necessarily explained by the training and test datasets (Figure 5 and Supplementary Figure S2). The annotation issues noted above could account for parts of the discrepancies, but there were also other features that may have caused the network to perform differently across years. The annotation of sandeel schools is easier for large schools (due to more stable frequency responses and higher signal to noise), and school size tends to increase with high sandeel abundance (Johnsen *et al.*, 2017). In years with low sandeel abundance, a higher proportion of small schools cause a more uncertain categorization. Furthermore, weather condition may affect the schooling behaviour, which affects sandeel school detection.

When reviewing model performance by year, especially when including all background pixels (Supplementary Figure S2), some of the discrepancies may have arisen due to erroneous annotation. For survey years up to 2008, the labelling tool was under development and labelling was less efficient, and typically square annotation boxes were used. A bug in the annotation software was discovered in 2013 that led to incorrect storing of the annotation information (but not the exported backscattering intensity values). This may explain the improvement in performance in later years. For 2015, the weather conditions were rough, which led to underestimated biomass as stated in the 2015–2017 survey reports. For 2016, the “0-group sandeel” class was introduced due to large amounts of juveniles, which indicates a change in the system that may cause the model to perform differently, or alternatively, caused the labelling to be more challenging. From this perspective, reviewing the performance of the model across years is an efficient tool to identify any potential biases in the data series, but these considerations also apply to our benchmark.

There are several future directions in which we would like to take this. Further improvements of the model could be to include net sampling data and depth as separate inputs, where net samples could provide additional species information, and due to the conical shape of the echosounder beam, range could be used to compensate for range effects (e.g. that schools at short range look different at longer ranges). We would also want to include the uncertainty of the acoustic categorizing to the survey abundance estimates and, consequently, the fisheries advice when fully automating the annotation process. Another particularly interesting property of CNNs is transfer learning, i.e. that a network can be initialized from a previously trained network, and when presented with new data can update its weights. When a network is developed for sandeel classification, we can apply transfer learning and

adapt the network for different species, ideally across a wide range of surveys.

We have shown that a CNN can be reliably trained to categorize acoustic multifrequency observations. The main strength of this method is that the parameters can be learned directly from the echosounder output using manual labels as training data, i.e. there is no need to predefine features like frequency response, school morphology, etc., as the network learns the features directly from the training data. The method also allows us to code the tacit “knowledge” of a skilled operator, and it would be interesting to see if the method could be used to replicate different operators. In conjunction with more traditional, physics-based methods, this would enable us to study drift in expert judgments, explain annual differences, etc. When the network is trained on other surveys, we can transfer networks between surveys and look for differences in practices and test the implications. In our opinion, an end-to-end training approach opens possibilities not achievable when using conventional methods.

Supplementary data

Supplementary material is available at the ICESJMS online version of the manuscript.

Acknowledgements

This work is a part of the COGMAR project, funded by the Norwegian Research Council (grant 270966).

References

- Fallon, N. G., Fielding, S., and Fernandes, P. G. 2016. Classification of Southern Ocean krill and icefish echoes using random forests. *ICES Journal of Marine Science*, 73: 1998–2008.
- Footo, K. G. 1983. Linearity of fisheries acoustics, with addition theorems. *The Journal of the Acoustical Society of America*, 73: 1932–1940.
- Footo, K. G., Knudsen, H. P., Vestnes, G., MacLennan, D. N., and Simmonds, E. J. 1987. Calibration of acoustic instruments for fish density estimation: a practical guide. ICES Cooperative Research Report 144.
- Freeman, S., Mackinson, S., and Flatt, R. 2004. Diel patterns in the habitat utilisation of sandeels revealed using integrated acoustic surveys. *Journal of Experimental Marine Biology and Ecology*, 305: 141–154.
- Furness, R. W. 2002. Management implications of interactions between fisheries and sandeel-dependent seabirds and seals in the North Sea. *ICES Journal of Marine Science*, 59: 261–269.
- Gastauer, S., Fässler, S. M. M., O'Donnell, C., Høines, Å., Jakobsen, J. A., Krysov, A. I., Smith, L., *et al.* 2016. The distribution of blue whiting west of the British Isles and Ireland. *Fisheries Research*, 183: 32–43.
- Gjøsæter, H., Bogstad, B., Tjelmeland, S., and Subbey, S. 2015. A retrospective evaluation of the Barents Sea capelin management advice. *Marine Biology Research*, 11: 135–143.
- Haralabous, J., and Georgakarakos, S. 1996. Artificial neural networks as a tool for species identification of fish schools. *ICES Journal of Marine Science*, 53: 173–180.
- Hariharan, B., Arbeláez, P., Girshick, R., and Malik, J. 2015. Hypercolumns for object segmentation and fine-grained localization. *In* 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 447–456.
- Horne, J. K. 2000. Acoustic approaches to remote species identification: a review. *Fisheries Oceanography*, 9: 356–371.
- ICES. 2016. Report of the Benchmark Workshop on Sandeel (WKSand 2016). ICES Document CM 2016/ACOM: 33.

- Ioffe, S., and Szegedy, C. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. <http://arxiv.org/abs/1502.03167> (last accessed 26 November 2019).
- Johnsen, E., Pedersen, R., and Ona, E. 2009. Size-dependent frequency response of sandeel schools. *ICES Journal of Marine Science*, 66: 1100–1105.
- Johnsen, E., Rieucan, G., Ona, E., and Skaret, G. 2017. Collective structures anchor massive schools of lesser sandeel to the seabed, increasing vulnerability to fishery. *Marine Ecology Progress Series*, 573: 229–236.
- Karp, W. A., and Walters, G. E. 1994. Survey assessment of semi-pelagic gadoids: the example of Walleye Pollock, *Theragra chalcogramma*, in the Eastern Bering Sea. *Marine Fisheries Review*, 56: 8–22.
- Kloser, R. J., Ryan, T., Sakov, P., Williams, A., and Koslow, J. A. 2002. Species identification in deep water using multiple acoustic frequencies. *Canadian Journal of Fisheries and Aquatic Sciences*, 59: 1065–1077.
- Korneliussen, R. J. (Ed). 2018. Acoustic target classification. ICES Cooperative Research Report 344. 104 pp.
- Korneliussen, R. J., Heggelund, Y., Macaulay, G. J., Patel, D., Johnsen, E., and Eliassen, I. K. 2016. Acoustic identification of marine species using a feature library. *Methods in Oceanography*, 17: 187–205.
- Korneliussen, R. J., and Ona, E. 2003. Synthetic echograms generated from the relative frequency response. *ICES Journal of Marine Science*, 60: 636–640.
- Long, J., Shelhamer, E., and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. *In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3431–3440. Boston, MA.
- Macer, C. T. 1966. Sand Eels (Ammodytidae) in the Southwestern North Sea: Their Biology and Fishery. *Fishery Investigations Series 2*, 24. H.M. Stationery Office, London.
- MacLennan, D. N., Fernandes, P. G., and Dalen, J. 2002. A consistent approach to definitions and symbols in fisheries acoustics. *ICES Journal of Marine Science*, 59: 365–369.
- Malde, K., Handegard, N. O., Eikvil, L., and Salberg, A.-B. 2020. Machine intelligence and the data-driven future of marine science. *ICES Journal of Marine Science*, 77: 1274–1285.
- Mukai, T., and Amakasu, K. 2016. Comparison of the volume backscattering strength measured by EK60 and EK80. *The Journal of the Acoustical Society of America*, 140: 3242–3242.
- Ona, E. 2003. An expanded target-strength relationship for herring. *ICES Journal of Marine Science*, 60: 493–499.
- Proud, R., Mangeni-Sande, R., Kayanda, R. J., Nyamweya, C., Ongore, C., Everson, I., and Elison, I. 2020. Acoustic identification of schools of the Silver Cyprinid *Rastrineobola argentea* in Lake Victoria using Random Forests. *ICES Journal of Marine Science*, 77: 1379–1390.
- Reid, D. G. 2000. Report on echo trace classification. ICES Cooperative Research Report 238. International Council for the Exploration of the Sea, Copenhagen, Denmark.
- Ronneberger, O., Fischer, P., and Brox, T. 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. <https://arxiv.org/abs/1505.04597> (last accessed 26 November 2019).
- Rumelhart, D. E., Hinton, G. E., and Williams, R. J. 1986. Learning representations by back-propagating errors. *Nature*, 323: 533–536.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., et al. 2015. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115: 211–252.
- Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., and LeCun, Y. 2013. OverFeat: Integrated Recognition, Localization and Detection using Convolutional Networks. <http://arxiv.org/abs/1312.6229> (last accessed 26 November 2019).
- Simmonds, J., and MacLennan, D. 2005. *Fisheries Acoustics: Theory and Practice*. Blackwell Science, Oxford. 437 pp.
- Trenkel, V. M., Mazauric, V., and Berger, L. 2008. The new fisheries multibeam echosounder ME70: description and expected contribution to fisheries research. *ICES Journal of Marine Science*, 65: 645–655.
- Winslade, P. 1974. Behavioural studies on the lesser sandeel *Ammodytes marinus* (Raitt) II. The effect of light intensity on activity. *Journal of Fish Biology*, 6: 577–586.
- Wuillez, M., Ressler, P. H., Wilson, C. D., and Horne, J. K. 2012. Multifrequency species classification of acoustic-trawl survey data using semi-supervised learning with class discovery. *The Journal of the Acoustical Society of America*, 131: EL184–EL190.
- Wright, P. J., Jensen, H., and Tuck, I. 2000. The influence of sediment type on the distribution of the lesser sandeel, *Ammodytes marinus*. *Journal of Sea Research*, 44: 243–256.

Handling editor: David Demer

Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Original Article

Highlighting growth regulation processes in fish populations by a simplex simulation approach: application to *Merluccius hubbsi* stocks in the Southwestern Atlantic

Nabil Semmar ^{1*} and André M. Vaz-dos-Santos ²

¹Laboratory of Bioinformatics, Biomathematics and Biostatistics, LR 16 IPT 09, Pasteur Institute of Tunis, University of Tunis El Manar, Tunis, Tunisia

²Sclerochronology Laboratory, Biodiversity Department, Federal University of Paraná, Rua Pioneiro, 2153 Jardim Dallas, CEP, Palotina, PR 85950-000, Brazil

*Corresponding author: tel: 00 216 20976197; e-mail: nabilsemmar5@gmail.com.

Semmar, N. and Vaz-dos-Santos, A. M. Highlighting growth regulation processes in fish populations by a simplex simulation approach: application to *Merluccius hubbsi* stocks in the Southwestern Atlantic. – ICES Journal of Marine Science, 77: 1401–1413.

Received 14 May 2019; revised 30 September 2019; accepted 13 November 2019; advance access publication 20 December 2019.

A new simplex-based simulation approach (*Spx*) was developed to highlight multidirectional and multi-scale relationships between morphometric variables helping to functionally differentiate biological (fish) groups for better stocks definition and monitoring. Application concerned *Merluccius hubbsi* sampled in 1968–1972 and 2004 in six Southwestern Atlantic areas. Simulation results highlighted negative trends opposing front to back compartments indicating competition for body biomass distribution. However, top and bottom parts within these compartments were positively correlated indicating cooperative processes in favour of target local growth regulation. Positive and negative trends of growth regulation were also highlighted at lower body scale, notably between smaller components constituting body front compartment. On a geographical scale, average regulation levels of same morphometric variables showed monotonic or alternated variations between successive fish groups. This highlighted target and modulated growth regulations governing biomass distribution in different body parts by geographical-dependent ways. Under dynamical aspect (1968–1972 vs. 2004), growth regulation of mouth tended to increase with time leading to conclude on morphometric responses of *M. hubbsi* to overfishing pressure. *Spx* results were confirmed by several traditional approaches which showed less integrative aspect.

Keywords: Huxley model, mixture design, morphometrics, simplex, simulation, stock identification

Introduction

Growth represents complex process associated with variable distributions of biomass at several biological scales extending from individual body to intra- and inter-population balance states. Growth states and mechanisms serve as key criteria to define groups with homogeneous vital rates (stock units in fishery management; Kerr *et al.*, 2017). The questions of identity, assessment, and monitoring of unit stocks represent classical aims in fishery management. Complexity of growth system is linked to

multifactorial, multidirectional, and multi-scale regulation trends between body components (morphometric variables) leading to specific aspects of different biological groups. Multiple aspects of growth in biological populations are classically approached by means of several complementary quantitative methods (Cadrin, 2000; Cadrin *et al.*, 2014; Secor, 2014):

Principal component analysis (PCA) is unsupervised method providing correlation charts between variables helping to highlight different trends associated with different groups (Semmar,

2011). Multiple correspondence analysis (MCA) highlights how different groups are associated with different variation ranges and ways of discrete variables (Escofier and Pagès, 1991). Discriminant analysis (DA) is supervised method used to identify/predict groups from quantitative profiles of measured (morphometric) variables (Seber, 1984). Linear and non-linear modelling methods provide relational aspects of growth governed by a limited number of significant variables (Seber, 1984). Body sizes and weights are generally used to model growth processes by traditional methods including Huxley and von-Bertalanffy models (Huxley, 1924, 1932; Weatherley and Gill, 1987; Weatherley, 1990; Saborido-Rey and Kjesbu, 2005).

Alternatively to several decomposition approaches, simulation methods provide integrative structured information on a studied population by exploring potentially its overall variability. Simulation methods helping to treat with multidimensional, multi-scale, and overlapping/mixing aspects of biological groups were encouraged in the recent years under the framework of management evaluation strategy (MES; Kerr et al. 2017). Integrative aspect can be satisfied by Markov Chain Monte Carlo sampling methods (Manly, 2007, Panikian et al., 2015). The current work presents a new simulation approach based on population stratification, simplex mixture design, and bootstrap sampling technique. Simplex rule is appropriate for mass distribution analysis between system constituents and through organization ways obeying to mass conservation law (Semmar and Roux, 2014). Stratification and bootstrap provided robust sampling ways for heteroscedasticity treatment and variability integration, respectively.

The new simulation simplex method (*Spx*) treats mixing aspects of biological groups constituting heterogeneous population and characterized by variability of morphometric profiles. *In silico* differential mixing and averaging of morphometric profiles helped to highlight growth regulation trends manifesting within and between groups. Mixing-based methodology of *Spx* is appropriate to treat the natural connections and overlapping aspect of biological groups.

Spx allowed fish stock identification, monitoring, and management from morphometric variability by responding to key questions: (i) “how length-expressed biomass was relatively distributed through different body parts conditionally to different fish groups considered a priori?” and (ii) “how lengths of different body parts can reflect specific growth ways of ecological populations in relation to intrinsic and/or external governing factors?”

Spx was applied on morphometric variables of the Argentine hake *Merluccius hubbsi* to analyse growth regulations of different body parts in fish groups associated with several geographical areas. Despite the overlapping aspects in natural populations, *Spx* was able to extract differentiation mechanisms specific to different groups helping for functional stock management.

Merluccius hubbsi Marini 1933 is one of the most important fishing resources in the Southwestern Atlantic Ocean, east coast of South America (Lloris et al., 2005). It is a demersal species continuously distributed from Brazil (21°S) to Argentina (55°S) in the continental shelf and slope, associated with water temperatures up to 23°C. In Brazilian waters (21°–34°S), two stocks (I and II) were defined (Southeastern, SE: 21°–29°S; Southern, S: 29°–34°S) and differentiated by several parameters including spawning’s areas and periods, first maturity, growth rate, otolith ring formation, otolith morphology, and young-of-the-year

development (Vaz-dos-Santos et al., 2009, 2017; Costa et al., 2018). Nevertheless, there is a southward gradient favouring dynamical exchanges between fish populations leading to mixing aspects of groups with unknown relative levels (Kerr et al., 2017). This leads to misleading risks that need to be overcome to avoid bias in stock identification, assessment, and management. *Spx* provided appropriate simulation approach to highlight how different groups can be functionally distinguished within their heterogeneous and naturally interconnected whole biosystem.

Spx was applied on two sampling periods of *M. hubbsi* (A: 1968–1972 vs. B: 2004). Six geographical sites were concerned (all in A vs. four in B) and were parts of the two conventional fish stocks (I and II). Traditional statistical analyses were performed on the two stocks (I and II) using log-transformed length data. Results showed limited differentiation between stocks. More integrative information was provided by *Spx* which used morphometric variables standardized by their sum (relative levels) to simulate growth regulation mechanisms governing variability and differentiation between and within population groups (stock units). This integrative approach is the first of its kind in fishing management field and it responds well to the need of MES development (Kerr et al., 2017).

Material and methods

Study area

The studied area concerned the Southwestern Atlantic Ocean between 21° and 34°S: Brazilian coast and its Economic Exclusive Zone (Figure 1). It included two defined stocks (I and II) associated with the Southeastern (21°–29°S) and Southern (29°–34°S) areas, respectively.

In the Southeast region, the cold-water mass “South Atlantic Central Water” (6°–20°C, 34–36 PSU) occupies the upper slope during all the year; during the austral spring to summer (October to March) it also moves to the continental shelf, remaining sub superficial (Piola et al., 2018). This is the main water mass in which *M. hubbsi* lives, although the species is also found in the

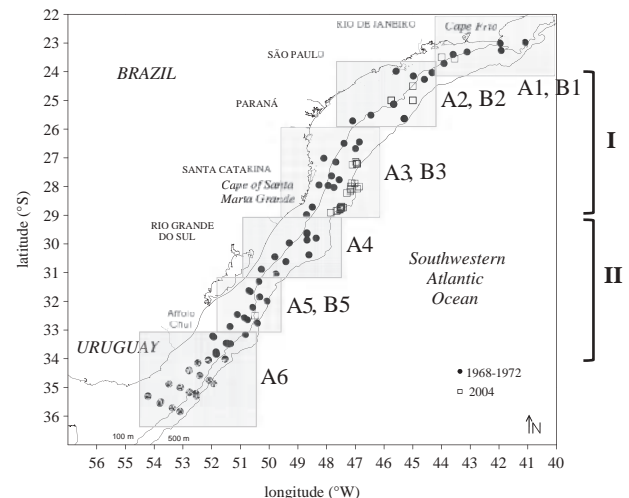


Figure 1. *Merluccius hubbsi*: sample sites in the two study periods 1968–1972 (A) and 2004 (B). Strata A_j and B_j were considered in *Spx* for simulations of growth regulation ways in different fish groups of periods A, B, I, II, conventionally defined stocks.

Shelf Water, a mixture of water masses close to the coast. The South region presents the same pattern, but during the austral autumn to winter (April to September), it experiences intrusions of the Sub-Antarctic Water (12°–15°C, 33 PSU; Piola *et al.*, 2018). Two important upwelling areas occur at 23°S (Cape Frio) and 28°40'S (Cape of Santa Marta Grande). In this second area, *M. hubbsi* does not spawn and juveniles do not occur (Vaz-dos-Santos and Schwingel, 2015).

Sampling and data acquisition

Merluccius hubbsi was sampled in two periods A (1968–1972) and B (2004; Figure 1) by reference to the scientific collection of Zoology Museum of São Paulo University (MZUSP) (A) and commercial landings (B). The specimens A (231 fishes) were obtained in bottom-trawl surveys performed in six areas (A1–A6) between 21° and 36°S along the continental shelf (up to 200 m isobaths; Supplementary material S1). The specimens B (516 fishes) were attained during January to February and July to August of 2004 from bottom-trawlers operating along the Brazilian coast (21°–34°S) in four areas (B1–B3, B5) with depths varying between 40 and 590 m (Supplementary material S1). The specimens were cleaned, fixed in formalin (10%), and stored in alcohol (70%) for ulterior analysis.

Morphometric data of *M. hubbsi* concerned 18 body measurements which were attained with a digital calliper (0.05 mm precision; Figure 2; Supplementary material S2; Inada, 1981; Lloris *et al.*, 2005). The specimens B were measured 2–3 d after sampling, avoiding shrinkage bias.

Traditional data analysis

Traditional computational analysis of growth processes based on morphometric data involved five steps detailed in Supplementary material S3 and summarized below (Cadrin, 2000):

- (1) Comparison of length frequency distributions between the two temporal populations A, B using three tests: Kolmogorov–Smirnov, Scheirer-Ray-Hare (non-parametric two-way analysis; an extension of Kruskal–Wallis test), Permutational Multivariate Analysis of Variance (PERMANOVA) (non-parametric permutation test to improve Multivariate Analysis of Variance (MANOVA);

- Supplementary material S3 and S4; Sokal and Rohlf, 1995; Anderson *et al.*, 2008).
- (2) Predictive modelling of the variation of 17 morphometric variables *Y* in relation to the total lengths (*X*) by applying 17 Huxley models ($Y = aX^b$) (Supplementary material S5; Huxley, 1924, 1932).
- (3) Correction of morphometric measures by using calculated parameter *b* and standard length *L*₀ (fixed at 210 mm): $Y' = Y(L_0/L)^b$ (Lombarte and Leonart, 1993).
- (4) Effect analysis of both fish stock and total length on the 17 morphometric variables by analysis of covariance (ANCOVA) (Sokal and Rohlf, 1995).
- (5) Application of DA for fish stocks (I and II) recognition from multivariate morphometric patterns (Gotelli and Ellison, 2004).

Simplex approach

Preliminary data processing

Spx is initially based on stratification of whole population into several groups characterized by different relative levels of system constitutive variables. Principle of *Spx* refers to mass conservation law where a whole resource can be distributed by multiple competing ways under unit sum constraint (sum of all distributed mass parts = 1, 100%) (Cornell, 2002). This conservative principle can be applied to measured lengths of different body parts, illustrated here by *M. hubbsi*.

Initially, size effect was removed by relativizing the different measured lengths by reference to their sum that was associated to a whole unit representing the perimeter of the covered body space. Figure 3a and b represents a generic fish with four body measurements combining front/back and top/bottom with illustrative values. Relativization of different body parts by reference to their perimeter (sum) leads to a morphometric profile with different segments varying relatively the ones at the expense of the others (Figure 3b and c). In the current study, two systems of morphometric variables standardized by their sum were considered: (i) *Pdd2*, *Lpa*, *Lbsd*, and *Lba* were considered to cover overall body perimeter of *M. hubbsi*. (ii) At lower morphometric scale, body front compartment was characterized by five constitutive variables: *Lbfd*, *Lpso*, *Ed*, *Lpo*, and *Lppv* (Figure 2).

Spx initially required population stratification into *q* groups representing *q* modalities or levels of a growth-influencing factor (Figure 3c). The effects of such a factor on growth are highlighted by smoothing group-dependent relationships between morphometric variables (Figure 3i). In this study, fish population was stratified by considering geographical sites and periods. This helped to analyse flexibility and stability of fish developments under spatial–temporal aspect. Initially, fish population was stratified into *q* = 6 and 4 groups sampled in periods A (A1–A6) and B (B1–B3, B5), respectively (Figure 1; Supplementary material S1). Two *Spx* were applied on A and B datasets, respectively.

Simulation methodological steps

Growth regulation processes governing length variation of body parts within and between *q* fish groups were simulated by *Spx* using a complete set of *N* combinations between groups (Figure 3e). The *N* combinations are given by a mixture design (Scheffé's matrix) containing *N* rows and *q* columns (Scheffé, 1958, 1963; Cornell, 2002, 2011). Each row *s* is associated with a specific mixture

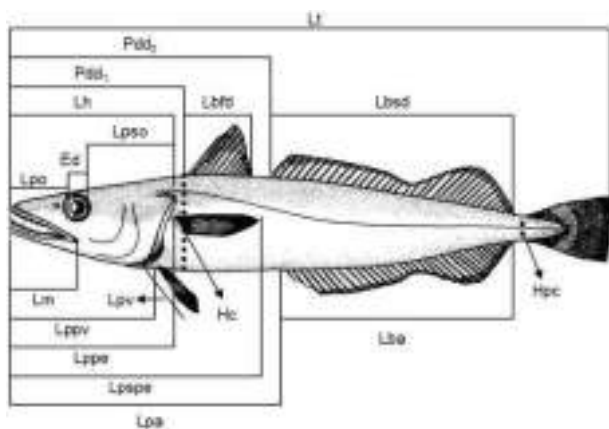


Figure 2. *Merluccius hubbsi*: body measurements for morphometric analyses.

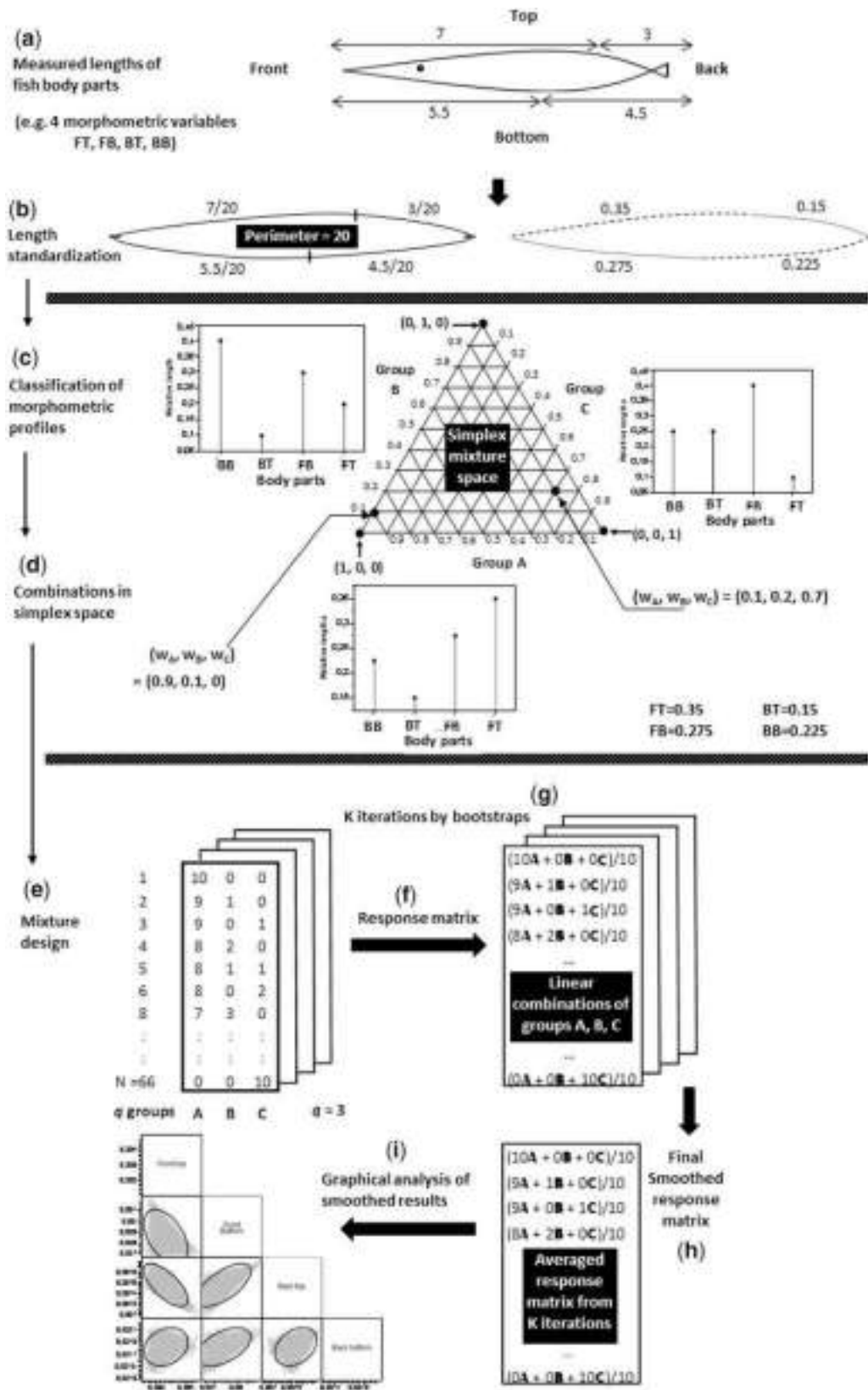


Figure 3. Different methodological steps of simulation simplex approach (illustrated here by $q=3$ groups).

combining the q groups j ($j = 1$ to q) with q weight values w_j (contributions) under the constraint of constant sum w :

$$\sum_{j=1}^q w_j = w \quad (1)$$

With $w_j = 0$ to w .

For instance, for $q = 3$, $(w_1 + w_2 + w_3) = (1 + 2 + 7)$ means a mixture made by 10%, 20%, and 70% of groups 1, 2, and 3, respectively (Figure 3d). The total number N of mixtures is initially given by a combinatorial formula depending on q and w :

$$N = C_{q-1}^{w+q-1} = \frac{(w+q-1)!}{(q-1)!w!} \quad (2)$$

With $w = 10$ contributing individuals per mixture of q groups, the formula gives $N = 286$ and 3003 mixtures with $q = 4$ and 6, respectively (associated with 4 and 6 fish groups in 2004 and 1968–1972, respectively).

Each mixture was applied by randomly sampling $w = 10$ individuals (fishes) from the q groups j by respecting the q weights w_j given by Scheffé's matrix. At the output of each mixture, the w contributing individuals representing the q weighted groups were averaged to calculate a barycentric profile containing average relative levels of morphometric variables. In all, a response matrix of N barycentric profiles was calculated (Figure 3f).

At this step, each barycentric profile was calculated from only w randomly sampled individuals whereas each population group contained much more individuals which did not contribute to mixtures leading to underestimation of variability within and between groups. This sampling deficit was solved by iterating the mixture design and its response matrix by bootstrap technique (Figure 3g; Manly, 2007): in all, $K = 30$ iterations were applied to the random sampling of $w = 10$ individuals for each mixture of Scheffé's matrix. This led to $K = 30$ elementary response matrices containing $N = 286$ and 3003 barycentric profiles in B and A , respectively. Finally, the 30 matrices were averaged to obtain a smoothed matrix integrating potentially the variability between and within groups (Figure 3h). From the smoothed matrix, regulation trends between morphometric variables were plotted (Figure 3i).

Geometrical highlighting of regulatory growth trends

Group-dependent relationships between paired morphometric variables were highlighted by projecting the weight values w_j (0–10) of group j on the corresponding points of simplex space. Points with equal weights were statistically covered by a 95% confidence ellipse. The succession of the $w + 1$ (=11) weight ellipses from 0 to $w = 10$ provided a trajectory indicating how the two considered morphometric variables varied the one relatively to the other in favour of the considered fish group (stock unit).

Comparison of *Spx* results to traditional methods

Checking, validation, and highlighting advantages of *Spx* results were performed by comparison with results from other methods (Supplementary material S7–S14).

Variation ranges analysis of growth regulation variables

Box plots of relative morphometric variables initially helped to characterize spatial–temporal fish groups by different variation ranges and levels of growth regulations.

Association analysis between geographical groups and growth regulation profiles

DA was applied to separate fish groups by their profiles of relative morphometric variables taken in single or combined forms. Attributions of growth regulation profiles to fish groups were carried out using Bayesian rule allocating a profile to the group showing maximal assignment probability (Seber, 1984).

Analysis of growth regulation trends and levels

Signs and levels of growth regulation trends in different groups were checked by several statistical methods including non-parametric Spearman correlations (*Spm*), correlation-based PCA and MCA (Escofier and Pagès, 1991; Semmar 2011, 2013):

Preliminary correlation analysis between relative levels of morphometric variables was invested by calculating *Spm* in different fish groups (Semmar, 2013). PCA was applied on relative levels of morphometric variables to obtain correlation charts highlighting characteristic regulation trends of groups (Semmar, 2011). MCA provides visualized trend analysis by considering variation levels of variables within fish groups. MCA was applied on a complete binary dataset where the variables were organized into four modalities corresponding to values less than first quartile Q1, between Q1 and median (Q2), between Q2 and third quartile (Q3), and more than Q3 (Escofier and Pagès, 1991).

Results

Traditional data analysis

Morphometric measures of total length are summarized in Table 1.

The size effect removal was necessary due to significant temporal variation in terms of total length ($p < 0.05$ in the three tests, Supplementary material S4). After size effect removal, data from both periods were joined ($F_{\text{PERMANOVA}} = 58.96$, $p = 0.629$) for the subsequent analyses.

Huxley models fitted to each stock did not show significant differences (Table 2), except for *Pdd2*, *Lpspe*, and *Lpa* due to the values of a coefficient (ANCOVA, $p < 0.05$).

For *Pdd2*, the values for each stock were $a_{SE} = 1.95178$ and $a_S = 1.94695$; for *Lpspe* $a_{SE} = 1.95217$ and $a_S = 1.94942$; for *Lpa* $a_{SE} = 1.95532$ and $a_S = 1.95128$, revealing longer morphometric measures of *M. hubbsi* in Southeastern stock (SE) than Southern stock (S).

Concerning b -coefficient, negative values revealed a decrease in the proportion of the measurements when Lt increased (Table 2). The models showed low level of multicollinearity (Variance Inflation Factors close to 1, the value of no correlation) and the residue analysis did not show any trend. Mann–Whitney test highlighted only *Pdd2* and *Lpa* as measurements capable to distinguish stocks after size effect removal.

The joint analysis of the measurements reinforced the difference between the SE and S stocks ($F_{\text{PERMANOVA}} = 3.843$, $p = 0.007$), although it was difficult to distinguish clear patterns. In the DA, the correct assignments after cross-validation presented an overall coincidence of 58.96%, with 62% for SE stock and 50% for S stock, revealing a limited power for stock identification based on body measurements.

Table 1. *Merluccius hubbsi*: summarized morphometric measures (Lt) in different stocks and years.

Years	Stock	Mean \pm SD (mm)	[Min, max] (mm)	Sample size
1968–1972	SE (21°–29°S)	166.34 \pm 68.60	74 \leq Lt \leq 352	59
	S (29°–34°S)	161.48 \pm 63.60	66 \leq Lt \leq 430	172
2004	SE (21°–29°S)	310.83 \pm 84.03	144 \leq Lt \leq 618	491
	S (29°–34°S)	374.60 \pm 73.20	246 \leq Lt \leq 501	25

Table 2. *Merluccius hubbsi*: summary of models (pooled) from analysis of covariance between log-data of body measurements and total length (Lt) of Southeastern and Southern stocks

Variables	ANCOVA										
	Constant		Lt			Stock			Mann–Whitney test		
	a	\pm s.e.	b	\pm s.e.	p-Value	Coefficient	\pm s.e.	p-Value	VIF	U	p-Value
Hc	1.47260	0.02250	–0.002710	0.00969	0.780	0.001040	0.002090	0.618	1.48	186531.0	0.8029
Hpc	0.87750	0.01920	0.005060	0.00821	0.538	–0.002180	0.001770	0.219	1.34	201298.0	0.2306
Lbfd	1.35480	0.01470	–0.000380	0.00626	0.952	0.000170	0.001350	0.901	1.35	204250.5	0.9529
Lbsd	1.91500	0.00794	–0.001770	0.00339	0.601	0.000755	0.000730	0.301	1.34	206671.0	0.4004
Lba	1.91773	0.00755	–0.000440	0.00322	0.892	0.000179	0.000695	0.796	1.34	207094.0	0.5918
Lh	1.75498	0.00765	0.000330	0.00326	0.920	–0.000134	0.000705	0.849	1.35	208294.5	0.2681
Pdd1	1.77879	0.00643	0.002090	0.00274	0.447	–0.000911	0.000593	0.125	1.35	202483.0	0.2562
Pdd2	1.94936	0.00605	–0.005700	0.00258	0.027	0.002415	0.000554	<0.001	1.34	208052.0	0.0005
Lpo	1.29500	0.01020	0.000540	0.00434	0.900	–0.000227	0.000938	0.809	1.35	206634.0	0.6407
Lm	1.45569	0.00906	–0.001760	0.00387	0.650	0.000762	0.000836	0.362	1.35	207861.5	0.3470
Lppv	1.71126	0.00899	–0.002990	0.00383	0.436	0.001258	0.000827	0.129	1.35	209239.0	0.0845
Lppe	1.75729	0.00741	–0.002830	0.00316	0.371	0.001197	0.000683	0.080	1.35	209325.5	0.1322
Lpspe	1.95080	0.00748	–0.003200	0.00319	0.317	0.001377	0.000690	0.046	1.34	206592.0	0.1883
Lpa	1.95330	0.00736	–0.004740	0.00314	0.132	0.002017	0.000679	0.003	1.35	201406.0	0.0210
Ed	1.07250	0.01510	–0.001080	0.00645	0.867	0.000470	0.001390	0.736	1.34	205779.0	0.8077
Lpso	1.42701	0.00995	–0.000480	0.00424	0.909	0.000198	0.000917	0.829	1.35	205605.0	0.7490
Lpv	1.45800	0.01160	–0.000490	0.00497	0.922	0.000220	0.001070	0.837	1.34	203838.5	0.6403

The cut-off probability for significance of model parameters was $p \leq 0.05$. In bold: significant values.

s.e., standard error of the coefficient; VIF, variance inflation factor; U, Mann–Whitney statistics, p-value, probability.

Spx-simulation results

Compared growth regulations between the two periods at overall population scale

Spx provided smoothed regulation trends between morphometric variables (Figure 4a and b). Initially, these trends were analysed at overall population scale including all the fish groups of period A vs. period B.

Period A showed larger variation ranges (diversification) than period B concerning growth regulations of perimeter body variables (*Pdd2*, *Lpa*, *Lbsd*, *Lba*; Figure 4a). This could indicate some perturbations in B due to overfishing activities. Periods A and B showed distinct variation spaces in (*Pdd2* vs. *Lpa*) and (*Pdd2* vs. *Lba*) leading to strong differentiation between the two temporal populations.

Positive trends concerned the pairs (*Pdd2*, *Lpa*) and (*Lbsd*, *Lba*) whereas negative trends opposed (*Pdd2*, *Lpa*) (front body variables) to (*Lba*, *Lbsd*) (back body variables), especially in A. Moreover, in B, *Lbsd* showed a loss of monotonicity towards the other variables.

At lower body scale, regulation trends between front body variables (*Lbfd*, *Lpso*, *Ed*, *Lpo*, *Lppv*) showed distinct variation spaces separating periods A and B (Figure 4b): higher regulation levels concerned *Lbfd*, *Ed* in A vs. *Lpo*, *Lpso*, *Lppv* in B. This highlighted significant increase of growth regulations of mouth part in B.

Functional differentiations between fish groups

Spx highlighted group-dependent trends making morphometric variables to reach different balance levels through different variation ways and relational shapes according to fish groups (Figure 5). The pairs (*Pdd2*, *Lpa*) and (*Lpso*, *Lppv*) were considered to illustrate such growth flexibility through body perimeter and within front compartment, respectively.

Pdd2 vs. *Lpa* showed positive global trends resulting from monotonic succession of different full-weight ellipses ($w = 10$) associated with different fish groups (Figure 5). This indicated that perimeter front variables (*Pdd2*, *Lpa*) were similarly influenced by environmental conditions associated with different geographical sites: growth regulation levels of both *Pdd2* and *Lpa* were high in groups A2, A4, B3, B5, low in A1, A6, B1, intermediate in A3, A5, B2. These different levels highlighted diversified/heterogeneous aspects within conventional stocks (I and II) due to different growth regulation mechanisms between neighbour fish groups (A1 vs. A2 vs. A3), (B1 vs. B2 vs. B3) (for stock I) and (A4 vs. A5 vs. A6) (for stock II). Such intra-stock diversity showed alternation of high and intermediate growth regulation levels in middle sites vs. minimal states in extreme sites (A1, B1, A6). Positive global trends were also highlighted for back variables (*Lbsd*, *Lba*) which were opposite to (*Pdd2*, *Lpa*) Supplementary material S6a, b, e, and f). This could be directly due to two different

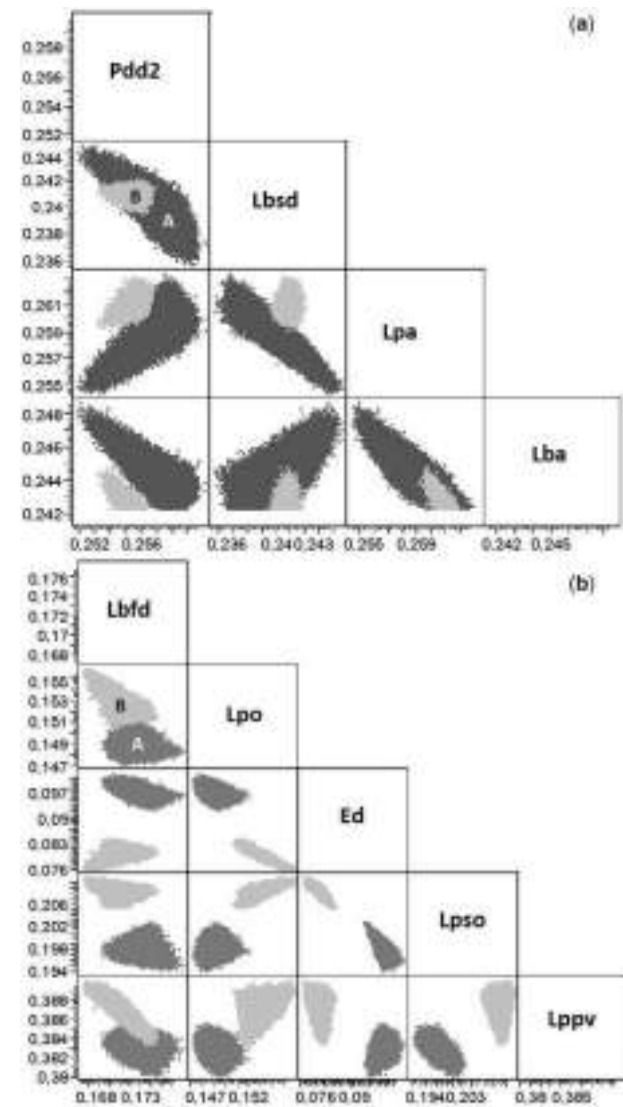


Figure 4. *Merluccius hubbsi*: smoothed trends between morphometric variables of body perimeter (a) and front part (b). Dark and light greys: periods A and B, respectively. Numbers of points in scatter plots = 30030 in A (10 × 3003) and 2860 in B (10 × 286) due to 10 replications of smoothed matrix.

mechanistic needs in fish: eating vs moving; it could also indicate opposite effects of environment on growth regulatory ways in body front and back compartments.

At local (within-group) scale, *Pdd2* vs. *Lpa* showed different inclinations of full-weight ellipses indicating flexible correlations between same variables (Figure 5): correlations were negative in A3, A4, A6, B5, slightly positive in A1, B1, and not significant in A2, A5, B2, B3. Negative local trends could be indicative of competition between top (*Pdd2*) and bottom (*Lpa*) body parts for biomass distribution in front compartment (A3, A4, A6, B5). However, positive trends indicated some cooperation mechanism between *Pdd2* and *Lpa* in favour of their common front compartment (in A1, B1). Local competing and local cooperative mechanisms were also highlighted for the pair (*Lbsd*, *Lba*) of back compartment with negative trends in A4, A6 vs. positive in A2, A5 (Supplementary material S6d and f vs S6b and e).

Further differentiations of fish groups were provided by analysing growth regulation trends between lower constituents of body front compartment (illustrated by *Lppv* vs. *Lpso*; Figure 6): in A, *Lpso* and *Lppv* showed strong negative trends at both global and local scales (Figure 6a–f). This indicated growth differentiation processes based on opposite regulations between top and bottom body segments. Extreme cases concerned A2 and A3 where (*Lpso*, *Lppv*) showed (maximal, minimal) and (minimal, maximal) states, respectively (Figure 6b and c). In B-fish groups, extreme global states concerned B3 and B5 with (minimal, maximal) and (maximal, maximal) regulation levels of (*Lpso*, *Lppv*), respectively (Figure 6i and j). Double maximum in B5 indicated a trend for bigger mouth development favoured by some environment conditions. However, locally, negative inclinations of full-weight ellipses indicated well-conserved intra-compartment competition for biomass distributions between top (*Lpso*) and bottom (*Lppv*) segments within the body front compartment (Figure 6j).

Spatial-temporal monitoring of average growth regulation states of stock units

Smoothed average states of body perimeter variables showed higher variability in period A than in B (Figure 7a). This could indicate higher resiliency of fish groups in A. Stock units were more differentiated by growth regulation levels than variation ranges: A4–A6 (stock II in 1968–1972) showed high *Lba* vs. low *Lpa* whereas B1–B3 (stock I in 2004) showed opposite aspect in favour of bigger mouth trend. A1–A3 (stock I in 1968–1972) had intermediate aspect with particularly extreme state in A2 (low *Lbsd*, *Lba* vs. high *Pdd2*, *Lpa*) making this fish group to be well distinguished. Opposition between A4–A6 and B1–B3 concerned also morphometric variables of front compartment (Figure 7b): high *Lbfd*, *Lpso*, *Lpo* vs. low *Ed* in B1–B3 with opposite trends in A4–A6. This highlighted clear effects of both geographical location and time to differentiate stock units.

Successive geographical areas resulted in target or modulated regulations of growths: for instance, in *Ed* vs. *Lbfd*, *Ed* increased from A1 to A6 whereas *Lbfd* showed alternated (cyclic) variations (Figure 7b). Target and modulated aspects were also highlighted for *Lba* vs. *Lbsd*, respectively (A2–A6; Figure 7a).

Comparison between Spx results and other methods

Variation ranges analysis of growth regulation variables

Box plots showed different average relative levels (growth regulations) of morphometric variables in different fish groups (Supplementary material S7; Table 3). Mann–Whitney test highlighted significant differences between the two periods A and B concerning *Lba*, *Ed* (higher in A), *Lpa*, *Lpso*, *Lpo*, *Lppv* (higher in B; $p < 10^{-4}$; Supplementary material S7). This agreed with *Spx* results showing clear variations of relative growth investments in fish groups with time (Figures 4 and 7). Also, box plots showed characteristic growth regulation levels of fish groups that agreed with *Spx* results (Table 3; Figure 7; Supplementary material S7): low *Lpa* (A1), low *Lbsd* vs. maximal *Lpa* (A2), maximal *Lbfd* vs. low *Lpo* (A3), maximal *Pdd2* vs. minimal *Lbsd* (A4), maximal *Ed* vs. minimal *Lpo* (A5), minimal *Pdd2*, *Lpa*, *Lpo* vs. maximal *Lbsd*, *Lba* (A6). In B, extreme growth regulation levels essentially concerned site B5 showing minimal *Lba*, *Lbfd*, *Ed* vs. maximal *Lpa*, *Lpso*, *Lpo*, *Lppv*.

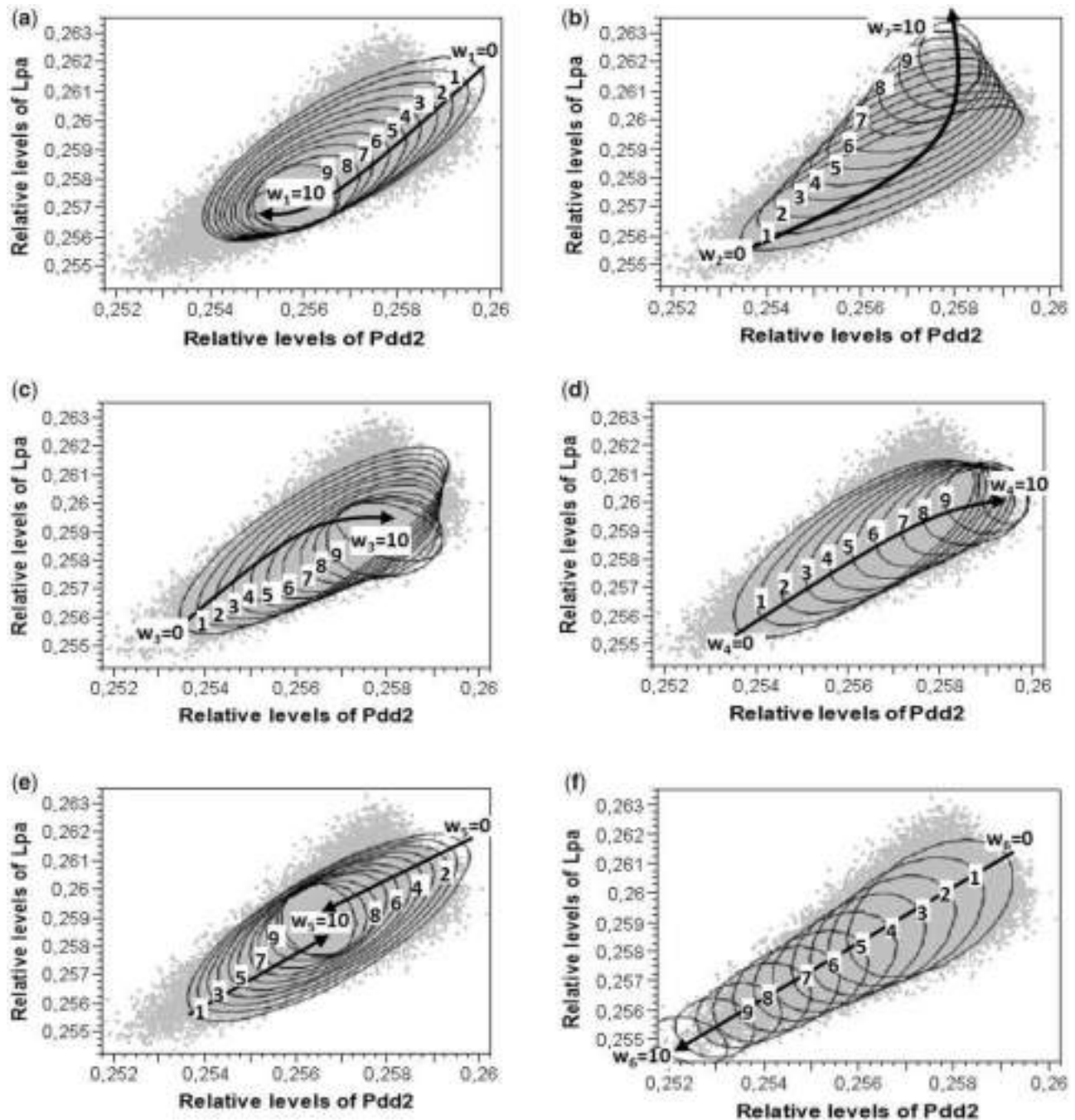


Figure 5. *Merluccius hubbsi*: smoothed plots showing different trends (bold trajectories) and local variations (weight ellipses' inclinations) governing regulations of morphometric variables *Lpa* and *Pdd2* in different fish groups associated with different geographical sites: A1 (a), A2 (b), A3 (c), A4 (d), A5 (e), A6 (f) in 1968–1972, B1 (g), B2 (h), B3 (i), and B5 (j) in 2004. The numbers of smoothed points in the scatter plots were equal to 30 030 in A (10×3003) and 2860 in B (10×286) due to 10 replications of smoothed matrix.

Association analysis between growth regulation profiles and geographical groups

DA used morphometric variables with single and/or combined (interactive) forms to provide predictive models of the ten fish groups (A1–A6, B1–B3, B5; [Supplementary material S8](#)). The percentages of corrected predictions varied between 65% (A5) and 83% (B5) ([Supplementary material S9](#)). Discriminant variables showed

extreme average levels in associated groups ([Supplementary material S7 and S9; Table 3](#)): A1, *Lbfd* (minimum average level); A2, *Lppv* (Min), *Lbfd* (high); A3, *Pdd2* (Max), *Lpso* (Min); A4, *Lbsd* (Min), *Lpso* (Low); A5, *Ed* (Max), *Lpso* (Min); A6, *Lba*, *Lbsd* (Max), *Lpa*, *Pdd2* (Min); B3, *Lpso*, *Lppv* (Max), *Lpa* (high), *Ed*, *Lba* (low); B5, *Lpo* (Max), *Ed* (Min). These variables showed extreme locations of full-weight ellipses of corresponding fish groups in *Spx* ([Figure 7](#)).

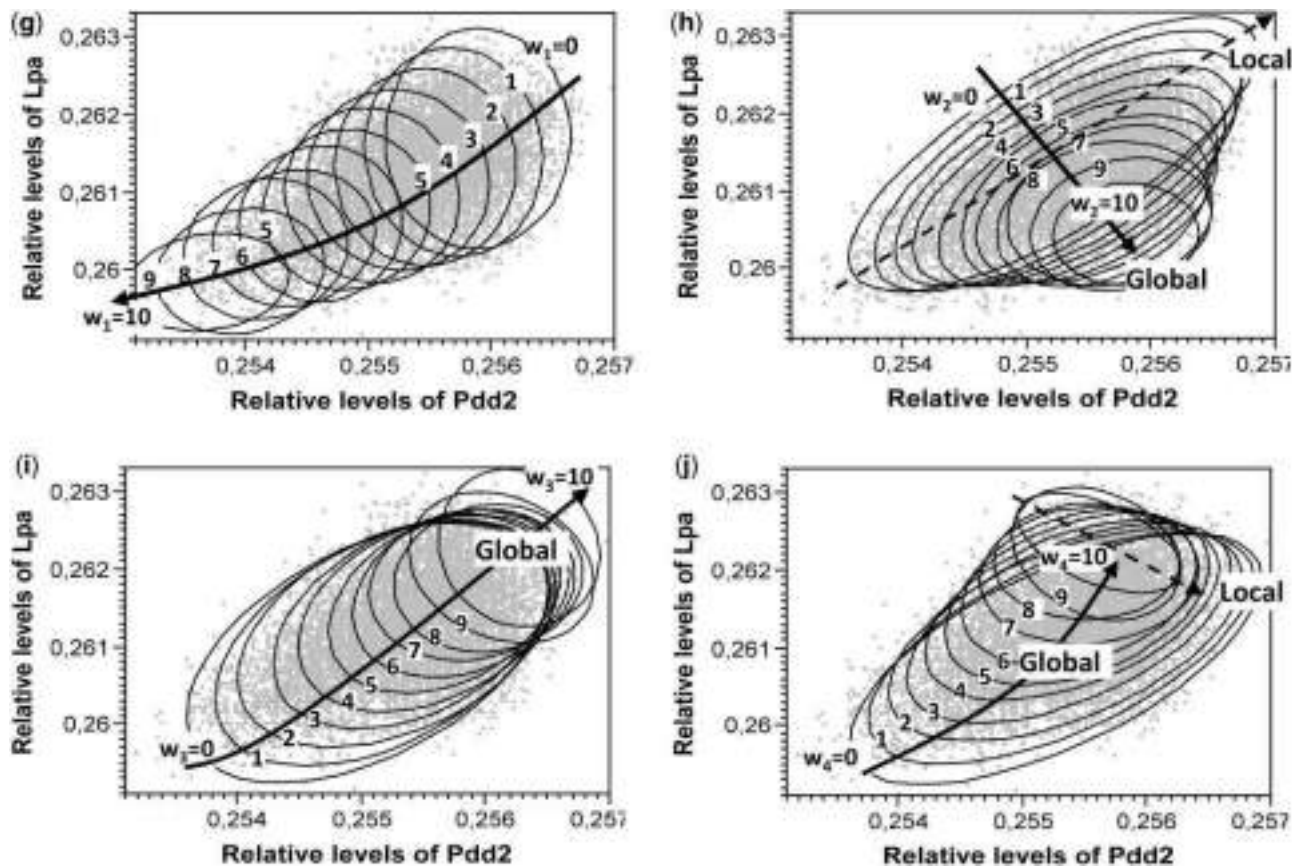


Figure 5. continued

Although DA models highlighted differentiation roles of morphometric variables, the development ways of groups remain unanswered question.

Analysis of trends between growth regulation variables

Spm negative correlations corresponded to negatively inclined full-weight ellipses in *Spx* (Supplementary material S10). Negative trends mainly concerned (*Lpa*, *Lbsd*), (*Pdd2*, *Lba*), (*Lbfd*, *Lppv*), (*Ed*, *Lpso*) in periods A and B followed by (*Pdd2*, *Lbsd*), (*Lpa*, *Lba*), (*Lpo*, *Lppv*), (*Lpso*, *Lppv*) in B (*Spm*) and both periods (*Spx*). These negative trends were also highlighted by PCA through opposite projections of variable points (Supplementary material S11): (*Lpa*, *Lbsd*) in A1, A3, A5, B3; (*Lba*, *Lpa*) in A2, B1, B2, B5; (*Pdd2*, *Lba*) in A1, A5; (*Ed*, *Lpso*) in A2, A5, A6, B1, B2, B3; (*Lppv*, *Lbfd*) in A2, A3, A6, B2, B3, B5; (*Pdd2*, *Lpa*) in A4.

Spm positive trends corresponded to positively inclined full-weight ellipses in *Spx* (Supplementary material S10). The most concerned pairwise was (*Lpo*, *Lpso*) in A2, A3, B1, B3, B5 (*Spx*) vs. B1, B2, B3 (*Spm*). It was followed by (*Ed*, *Lppv*) in A3 (*Spx*) vs. A3, B1, B3 (*Spm*), (*Lpso*, *Lbfd*) in A1, A2, A4 (*Spx*) vs. A4 (*Spm*), (*Pdd2*, *Lpa*) in A1, B1 (*Spx*). These positive correlations were confirmed in PCA by variable points showing close projections in factorial plots (Supplementary material S11).

Also, *Spx* and *Spm* agreed concerning not detected (not significant) trends indicated by high *p*-values in *Spm* and by circular or

not inclined full-weight ellipses in *Spx* (Supplementary material S10): e.g. *Lba* vs. *Lbsd* in A1, A3, B1–B3, B5 (Supplementary material S6a, c, and g–j). This could reveal non-linear dependences between the variables.

Compared with *Spm* and PCA, *Spx* advantageously provided visualization of variation spaces of group-dependent trends. Such group-depending associations between correlation signs and variation ranges of variables were checked by MCA leading to further validation of *Spx* results.

Multiple correspondence analysis

MCA highlighted strong separation between fish groups of periods A and B due to differences in growth regulation of front body variables (Supplementary material S12a). Period B showed high regulations of *Lpso* and *Lpo* (*Lpso4*, *Lpo4*) contrary to *Ed* (*Ed1*) vs. opposite aspect in A (*Lpso1*, *Lpo1*, *Ed4*; Supplementary material S12b). These results agreed with those of *Spx* which highlighted strong separation between groups A1–A6 and B1–B3, B5 within the variation space of balance average regulations *Lpso* vs. *Ed* (Supplementary material S12c and d).

MCA applied on body perimeter variables (*Pdd2*, *Lpa*, *Lbsd*, *Lba*) highlighted clear differentiations between fish groups A1–A6 (Supplementary material S13a and b). Results highlighted extreme states of A6, A4, and A2: high regulations of *Lba*, *Lbsd* vs. low levels of *Lpa*, *Pdd2* in A6 (*Lbsd4*, *Lba4*, *Lpa1*, *Pdd2.1*); high

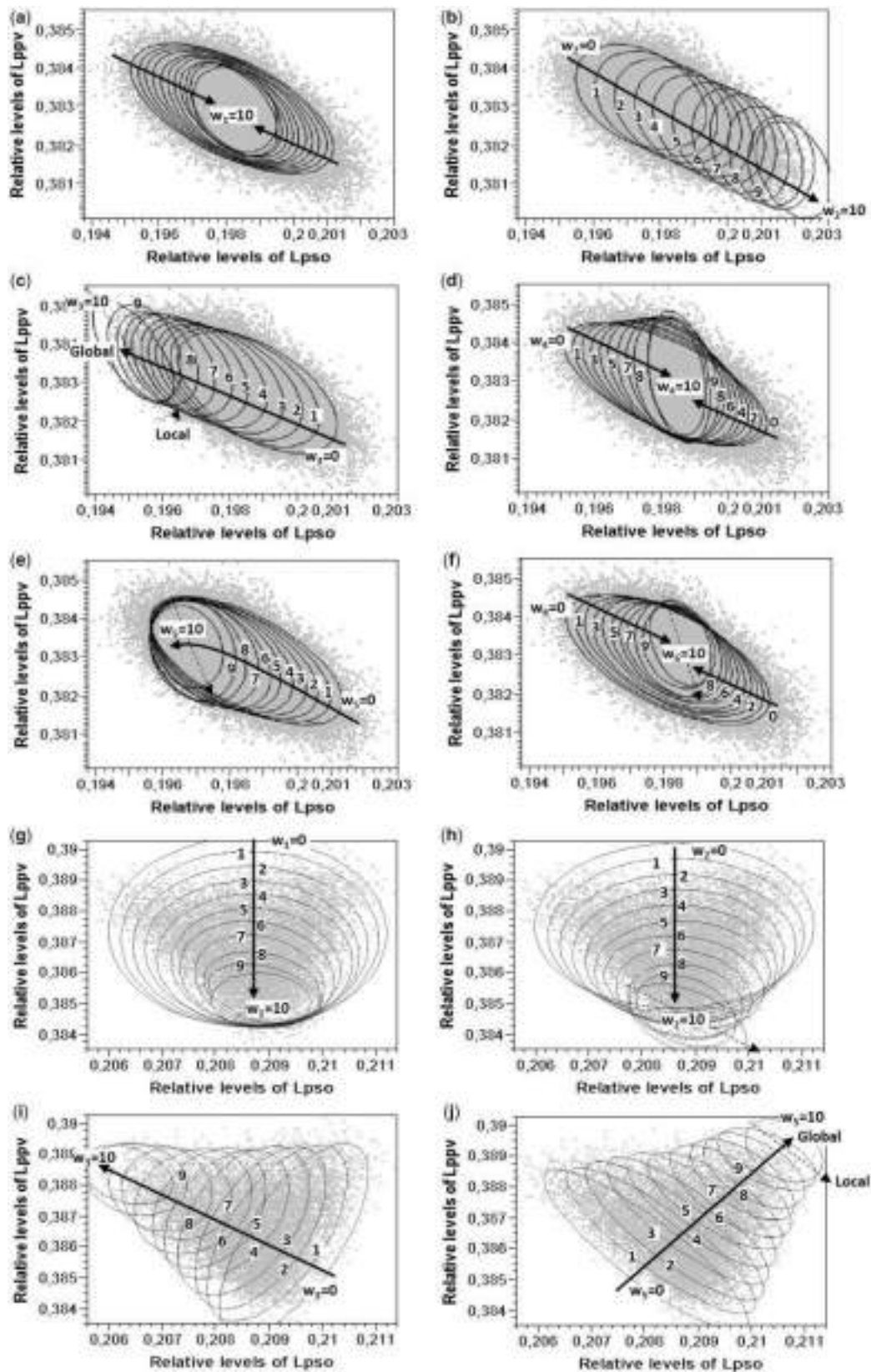


Figure 6. *Merluccius hubbsi*: smoothed plots showing different trends (bold trajectories) and local variations (weight ellipses' inclinations) governing regulations of morphometric variables *Lppv* and *Lpso* in different fish groups associated with different geographical sites A1 (a), A2 (b), A3 (c), A4 (d), A5 (e), A6 (f) in 1968–1972, B1 (g), B2 (h), B3 (i), and B5 (j) in 2004. The numbers of smoothed points in the scatter plots were equal to 30 030 in A (10×3003) and 2860 in B (10×286) due to 10 replications of smoothed matrix.

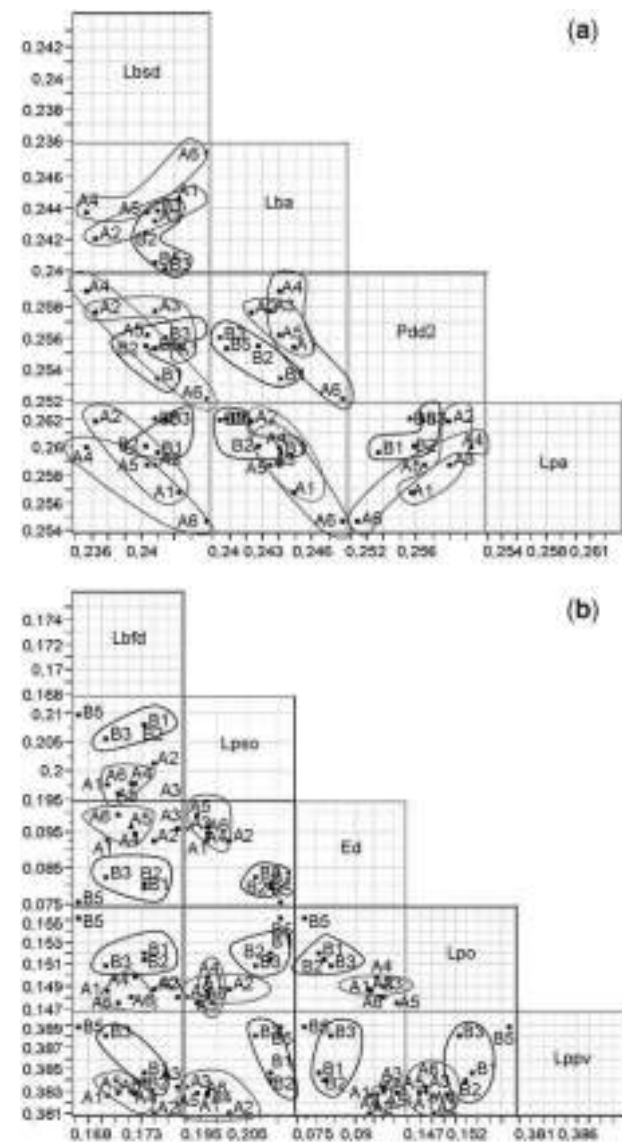


Figure 7. *Merluccius hubbsi*: variation of average states of morphometric variables corresponding to full-weight ellipses in different smoothed relationships by *Spx*. (a) Body perimeter variables and (b) front compartment variables. A1–A6, fish groups of period A (1968–1972); B1–B3 and B5, fish groups of period B (2004).

regulations of *Lpa*, *Pdd2* vs. low levels of *Lbsd* in A4, A2 (*Lpa4*, *Pdd2.4*, *Lbsd1*) and *Lba* in A2 (*Lba1*). A1 showed intermediate states between A6 and A2, A4 (*Pdd2.2*, *Pdd2.3*, *Lbsd2*, *Lbsd3*, *Lba3*, *Lpa3*). All these results agreed with those of *Spx* (Supplementary material S13c). Concerning intermediate states of A3, A5 highlighted by *Spx*, they were revealed by MCA by widely dispersed points along F1 for A5 and F2 for A3. ACM results highlighted some heterogeneity aspects of A3, A5 compared with other groups.

MCA applied on constitutive parts of front compartment (*Lbfd*, *Lpso*, *Lppv*, *Lpo*, *Ed*) highlighted differentiations between the four fish groups (B1–B3, B5) in 2004 (Supplementary material S14a and b). The four modalities of different variables were projected in different factorial subspaces that were associated with

different fish groups: *Lpso* and *Lpo* increased from B3 to B5 via B2 and B1, respectively; *Ed* showed opposite trend. Regulation levels of *Lbfd* increased from B5 (*Lbfd1*) to B1 (*Lbfd4*) via B3 and B2, respectively; *Lppv* showed opposite trend. These results were highlighted by *Spx* which also provided the balance levels of variables the ones relatively to the others (Supplementary material S14c).

Discussion

Spx provided an original simulation method of growth regulation trends between body constitutive variables helping to understand better mechanistic origins of population diversification and differentiation ways of biological groups. Based on population stratification and groups' weighting (combinations), *Spx* has the advantage to treat heterogeneous systems containing several groups characterized by unlimited number of variables whatever their variability levels could be.

Spx highlighted different scales mechanisms governing complex growth variability between and within several fish population groups: global trends governed growth regulation levels and differentiations between groups (under different overall/environmental conditions). However, local trends highlighted intrinsic growth regulation ways within groups leading to competing or cooperating body parts for local biomass distribution. By this way, *Spx* provides mechanistic information helping to define/identify stocks with specific growth regulation ways of different body parts. This provides highlighting of characteristic processes responsible for self-sustainability and resiliency of population groups in responses to environmental conditions and fishing activities.

Six fish groups coming from six geographical areas in 1968–1972 were well discriminated by well distinct regulation levels of body perimeter variables including *Pdd2*, *Lpa* in front compartment and *Lbsd*, *Lba* in back compartment (Figure 5; Supplementary material S6a and c). This indicated a strong order of biomass distribution in fish groups at whole body scale due to intense growth rates of juveniles (Vaz-dos-Santos and Rossi-Wongtschowski, 2007; Costa et al., 2018). However, in 2004, regulation ratios among most hierarchical variables (*Pdd2*, *Lpa*, *Lbsd*, *Lba*) showed narrower variation ranges indicating more homogeneous growth of adults (Figure 4a). This could result from lower intraspecific competition for foraging due to overfishing: overfishing leads to a reduction of stock abundance favouring relatively high prey abundance and subsequently higher capture per individual predator (*M. hubbsi*) (Perez et al., 2009; Vaz-dos-Santos et al., 2010). Moreover, higher and lower variability in A and B, respectively, seemed to be associated with two development strategies in fish populations (Minto et al., 2008; Panikian et al., 2015): higher resiliency in A vs. higher viability in B due to overfishing resulting in lower intraspecific competition for feeding.

At low body scale, *Lpso*, *Lppv*, and *Lpo* showed frankly higher regulation levels in 2004 than 1968–1972 indicating strong positive trend within body front compartment during the three decades (Figure 7b). Considering the relatively higher food availability (prey abundances) in 2004 (Muto and Soares, 2011), bigger jaws could be developed for higher capture-yielding in *M. hubbsi* vs. a decrease of relative growth of back body part (*Lba*; less required moving for prey capture; Figure 7a and b). This provides mechanistic argument on the fact that overexploited fishes tend to grow faster with target biomass distribution in body (Hart and Reynolds, 2002).

Table 3. *Merluccius hubbsi*: means and standard deviations of regulation levels of different morphometric variables of body perimeter (*Lbsd*, *Lba*, *Pdd2*, *Lpa*) and front compartment (*Lbfd*, *Lpso*, *Ed*, *Lpo*, *Lppv*) in fish groups of periods A (A1–A6) and B (B1–B3, B5).

Strata	<i>Lbsd</i>	<i>Lba</i>	<i>Pdd2</i>	<i>Lpa</i>	<i>Lbfd</i>	<i>Lpso</i>	<i>Ed</i>	<i>Lpo</i>	<i>Lppv</i>
A	0.241 ± 0.009	0.246 ± 0.008	0.256 ± 0.007	0.257 ± 0.009	0.172 ± 0.011	0.198 ± 0.009	0.098 ± 0.011	0.149 ± 0.008	0.383 ± 0.012
B	0.241 ± 0.006	0.243 ± 0.006	0.255 ± 0.005	0.261 ± 0.007	0.172 ± 0.009	0.208 ± 0.008	0.082 ± 0.008	0.152 ± 0.005	0.386 ± 0.01
A1	0.242 ± 0.008	0.244 ± 0.006	0.256 ± 0.006	0.258 ± 0.007	0.17 ± 0.012	0.198 ± 0.011	0.099 ± 0.012	0.149 ± 0.005	0.383 ± 0.011
A2	0.237 ± 0.007	0.243 ± 0.008	0.258 ± 0.006	0.262 ± 0.012	0.174 ± 0.011	0.202 ± 0.009	0.094 ± 0.012	0.149 ± 0.009	0.381 ± 0.013
A3	0.240 ± 0.009	0.244 ± 0.008	0.258 ± 0.01	0.258 ± 0.009	0.176 ± 0.014	0.195 ± 0.01	0.097 ± 0.009	0.148 ± 0.006	0.384 ± 0.011
A4	0.236 ± 0.01	0.244 ± 0.008	0.259 ± 0.006	0.261 ± 0.009	0.172 ± 0.012	0.198 ± 0.008	0.096 ± 0.011	0.150 ± 0.015	0.383 ± 0.016
A5	0.240 ± 0.009	0.244 ± 0.008	0.257 ± 0.006	0.259 ± 0.009	0.171 ± 0.011	0.197 ± 0.012	0.101 ± 0.012	0.148 ± 0.006	0.383 ± 0.012
A6	0.244 ± 0.007	0.248 ± 0.007	0.253 ± 0.007	0.255 ± 0.008	0.172 ± 0.01	0.198 ± 0.008	0.098 ± 0.011	0.148 ± 0.005	0.383 ± 0.01
B1	0.241 ± 0.006	0.244 ± 0.006	0.254 ± 0.005	0.26 ± 0.007	0.173 ± 0.009	0.209 ± 0.008	0.08 ± 0.008	0.152 ± 0.005	0.385 ± 0.009
B2	0.240 ± 0.006	0.243 ± 0.006	0.256 ± 0.006	0.261 ± 0.007	0.173 ± 0.01	0.209 ± 0.007	0.082 ± 0.008	0.152 ± 0.005	0.384 ± 0.009
B3	0.241 ± 0.006	0.241 ± 0.006	0.256 ± 0.005	0.262 ± 0.008	0.170 ± 0.009	0.206 ± 0.009	0.084 ± 0.008	0.151 ± 0.006	0.388 ± 0.01
B5	0.241 ± 0.004	0.240 ± 0.005	0.256 ± 0.004	0.264 ± 0.006	0.168 ± 0.009	0.211 ± 0.007	0.077 ± 0.005	0.155 ± 0.005	0.389 ± 0.01

Black and grey highlighting indicated maximal and minimal levels of different systems.

Spx results were compatible with regional hydrological phenomena: in the period of 1968–1972, A3 was distinct from the Southeast A1 and South extremes (A5, A6) by extreme growth regulations of front body parts (more particularly *Lbfd*; Figure 7b). This result reinforced the key role of the upwelling systems and the Patos-Mirim and La Plata systems (Piola et al., 2018). Moreover, the sequence and closeness of A4, A5, and A6 confirms that *M. hubbsi* is shared with Uruguay and Argentina (Vaz-dos-Santos et al., 2017).

Spx shows several methodological strengths and application perspectives for stock identification, monitoring, and management:

Fish groups can be openly characterized by regulation trends of several types of constitutive body variables including length, weight, feeding, body composition, otoliths, etc. Moreover, population stratification admits flexible criteria that can be intrinsic (physiological states, maturity degrees, ages, body length ranges, population sizes, stock biomasses) or extrinsic (environmental conditions, fishing activities). Therefore, simulated growth regulation trends could provide helpful information on gradual, transitory, or chained mechanisms implied in variability, persistency, adaptive strategies, and/or stock recruitment relationships (Minto et al., 2008; Panikian et al., 2015). However, higher number of stratification groups *q* can be limiting because of resulting drastic increase of total number of combinations (*N*). Although *Spx* does not provide predictive formulation, it has the ultimate advantage to highlight multidirectional, multi-shape, and multi-scale trends (group-dependent trends) preparing for next predictive modelling between regulation variables.

Application perspectives of *Spx* extend beyond fishpopulations. Among modern approaches, DNA barcoding provides reliable background reference libraries from which unidentified states or specimens can be delineated, classified, or precisely recognized (Hubert and Hanner, 2015). DNA barcodes can be used as categorical variables for population stratification in *Spx*. This helps to highlight regulation trends associating genomic categories of fish populations with different body constitution patterns (length, weight, feeding, metabolism, etc.).

Critics were given about working with preserved and fixed specimens leading to shrinkage bias on morphometric measures. Such a bias is strongly reduced by *Spx* because of three methodological factors: (i) initial working on ratio variables followed by (ii) pattern averaging then (iii) iteration of complete set of average patterns. Working on ratio variables (*X/Y*) leads to variation space with considerably lower variance and coefficient of variation compared to measured variable *X* (Supplementary material S15). Using morphometric data initially increased with random errors of 5–10%, simulation revealed stable regulation trends indicating strong attenuation of shrinkage bias with relative variables (Supplementary Material S16). Finally, mathematical demonstration based on error theory was given on bias reduction associated with ratio variable (Supplementary Material S17).

Supplementary data

Supplementary material is available at the ICESJMS online version of the manuscript.

Acknowledgements

NS and AMVS thank the Brazilian National Council for the Scientific and Technological Development (CNPq) for the financial support (Process 453871/2016-0) and research grant (Process 310451/2018-3). AMVS is grateful to Professors

Carmen Lúcia Del Bianco Rossi-Wongtschowski, José Lima de Figueiredo, Naércio Aquino Menezes, and Francisco Manoel de Souza Braga, whose discussions on the theme in the middle of the 2000s were valuable.

References

- Anderson, M. J., Gorley, R. N., and Clarke, K. R. 2008. PERMANOVA+ for PRIMER: Guide to Software and Statistical Methods. PRIMER-E Ltd, Plymouth.
- Cadrin, S. X. 2000. Advances in morphometric identification of fishery stocks. *Reviews in Fish Biology and Fisheries*, 10: 91–112.
- Cadrin, S. X., Kerr, L. A., and Mariani, S. (Eds). 2014. *Stock Identification Methods: Applications in Fishery Science*, 2nd edn. Elsevier, Oxford. 588 pp.
- Cornell, J. A. 2002. *Experiments with Mixtures: Designs, Models and the Analysis of Mixture Data*, 2nd edn. John Wiley & Son, New York. 649 pp.
- Cornell, J. A. 2011. A retrospective view of mixture experiments. *Quality Engineering*, 23: 315–331.
- Costa, P. A. S., Braga, A., da, C., Vieira, J. M., da S., Martins, R. R. M., São-Clemente, R. R. B. de, and Couto, B. R. 2018. Age estimation, growth and maturity of the Argentine hake (*Merluccius hubbsi* Marini, 1933) along the northernmost limit of its distribution in the south-western Atlantic. *Marine Biology Research*, 14: 728–738.
- Escofier, B., and Pagès, J. 1991. Presentation of correspondence analysis and multiple correspondence analysis with the help of examples. *In Applied Multivariate Analysis in SAR and Environmental Studies*, pp. 1–32. Ed. by J. Devillers and W. Karcher. Kluwer Academic Publishers, Dordrecht.
- Gotelli, N. J., and Ellison, A. M. 2004. *A Primer of Ecological Statistics*. Sinauer Associates, Sunderland. 510 pp.
- Hart, P. J. B., and Reynolds, J. D. 2002. *Handbook of Fish Biology and Fisheries, Volume 1: Fish Biology*. Blackwell Publishing, Malden. 413 pp.
- Hubert, N., and Hanner, R. 2015. DNA barcoding, species delineation and taxonomy: a historical perspective. *DNA Barcodes*, 3: 44–58.
- Huxley, J. S. 1924. Constant differential growth-ratios and their significance. *Nature*, 114: 895–896.
- Huxley, J. S. 1932. *Problems of Relative Growth*. Methuen, London. 319 pp.
- Inada, T. 1981. Studies on the merlucciid fishes. *Bulletin of the Far Seas Fisheries Research Laboratory*, 18: 1–172.
- Kerr, L. A., Hintzen, N. T., Cadrin, S. X., Clausen, L. W., Dickey-Collas, M., Goethel, D. R., Hatfield, E. M. C. *et al.* 2017. Lessons learned from practical approaches to reconcile mismatches between biological population structure and stock units of marine fish. *ICES Journal of Marine Science*, 74: 1708–1722.
- Lloris, D., Matallanas, J., and Oliver, P. 2005. Hakes of the World (Family Merlucciidae). An Annotated and Illustrated Catalogue of Hake Species Know to Date. FAO, Rome. 57 pp.
- Lombarte, A., and Leonart, J. 1993. Otolith size changes related with body growth, habitat depth and temperature. *Environmental Biology of Fishes*, 37: 297–306.
- Manly, B. F. J. 2007. *Randomization, Bootstrap and Monte Carlo Methods in Biology*. Chapman & Hall, Boca Raton. 447 pp.
- Minto, C., Myers, R. A., and Blanchard, W. 2008. Survival variability and population density in fish populations. *Nature*, 452: 344–348.
- Muto, E. Y., and Soares, L. S. H. 2011. Spatio-temporal variations in the diet and stable isotope composition of the Argentine hake *Merluccius hubbsi* Marini, 1933 of the continental shelf of Southeastern Brazil. *Marine Biology*, 158: 1619–1630.
- Panikian, G., Cussens, J., and Pitchford, W. 2015. Identification and quantification of heteroscedasticity in stock-recruitment relationships. *Canadian Journal of Fisheries and Aquatic Sciences*, 72: 1259–1271.
- Perez, J. A., Pezzuto, P. R., Wahrlich, R., and Soares, A. L. S. 2009. Deep-water fisheries in Brazil: history, status and perspectives. *Latin American Journal of Aquatic Research*, 37: 513–542.
- Piola, A. R., Palma, E. D., Bianchi, A. A., Castro, B. M., Dottori, M., Guerrero, R. A., Marrari, M. *et al.* 2018. Physical oceanography of the SW Atlantic Shelf: a review. *In Plankton Ecology of the Southwestern Atlantic*, pp. 37–56. Ed. by M. S. Hoffmeyer, M. E. Sabatini, F. P. Brandini, D. L. Calliari, and N. H. Santinelli, Springer, Cham. 586 pp.
- Saborido-Rey, F., and Kjesbu, O. 2005. Growth and maturation dynamics. Digital CSIC Paper, 1–26. <http://hdl.handle.net/10261/47150> (last accessed 4 December 2019).
- Scheffé, H. 1958. Experiments with mixtures. *Journal of the Royal Statistical Society: Series B*, 20: 344–360.
- Scheffé, H. 1963. The simplex-centroid design for experiments with mixtures. *Journal of the Royal Statistical Society: Series B*, 25: 235–263.
- Seber, G. A. F. 1984. *Multivariate Observations*. Wiley, New York. 686 pp.
- Secor, D. H. 2014. The unit stock concept: bounded fish and fisheries. *In Stock Identification Methods: Applications in Fishery Science*, 2nd edn, pp. 7–28. Ed. by S. X. Cadrin, L. A. Kerr, and S. Mariani, Elsevier, Oxford. 588 pp.
- Semmar, N. 2011. *Computational Metabolomics*. Nova Science Publishers, New York. 238 pp.
- Semmar, N. 2013. *Native Statistics for Natural Sciences*. Nova Science Publishers, New York. 515 pp.
- Semmar, N., and Roux, M. 2014. A new simplex approach to highlight multi-scale feeding behaviors in forager species from stomach contents: application to insectivore lizard population. *Biosystems*, 118: 60–75.
- Sokal, R. R., and Rohlf, F. J. 1995. *Biometry: The Principles and Practice of Statistics in Biological Research*, 3rd edn. W.H. Freeman and Company, New York. 887 pp.
- Vaz-dos-Santos, A. M., and Rossi-Wongtschowski, C. L. D. B. 2007. Age and growth of the Argentine hake *Merluccius hubbsi* Marini, 1933 in the Brazilian South-Southeast Region during 1996–2001. *Neotropical Ichthyology*, 5: 375–386.
- Vaz-dos-Santos, A. M., Rossi-Wongtschowski, C. L. D. B., and Figueiredo, J. L. 2009. *Merluccius hubbsi* (Teleostei: Merlucciidae): stock identification based on reproductive biology in the South-Southeast Brazilian region. *Brazilian Journal of Oceanography*, 57: 17–31.
- Vaz-dos-Santos, A. M., Rossi-Wongtschowski, C. L. D. B., Figueiredo, J. L., and Ávila-da-Silva, A. O. 2010. Threatened fishes of the world: *Merluccius hubbsi* Marini, 1933 (Merlucciidae). *Environmental Biology of Fishes*, 87: 349–350.
- Vaz-dos-Santos, A. M., and Schwingel, P. R. 2015. Biology and fisheries of hake (*Merluccius hubbsi*) in Brazilian waters, Southwest Atlantic Ocean. *In Hakes: Biology and Exploitation*, pp. 211–233. Ed. by H. Arancibia. John Wiley and Sons, Chichester, West Sussex. 348 pp.
- Vaz-dos-Santos, A. M., Santos-Cruz, N. N., Souza, D., Silva, A. G., Gris, B., and Rossi-Wongtschowski, C. L. D. B. 2017. Otoliths sagittae of *Merluccius hubbsi*: an efficient tool for the differentiation of stocks in the Southwestern Atlantic. *Brazilian Journal of Oceanography*, 65: 520–525.
- Weatherley, A. H. 1990. Approaches to understanding fish growth. *Transactions of the American Fisheries Society*, 119: 662–672.
- Weatherley, A. H., and Gill, H. S. 1987. *The Biology of Fish Growth*. Academic Press, London. 443 pp.



Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Original Article

Natural mortality estimation using tree-based ensemble learning models

Chanjuan Liu^{1,2,3}, Shijie Zhou^{2*}, You-Gan Wang³, and Zhihua Hu¹

¹School of Economics & Management, Shanghai Maritime University, Shanghai 201306, China

²CSIRO Oceans and Atmosphere, 306 Carmody Road, St Lucia, Brisbane, QLD 4067, Australia

³School of Mathematical Sciences, Science and Engineering Faculty, Queensland University of Technology, Brisbane, QLD 4001, Australia

*Corresponding author: tel: + 61 7 38335968; fax: + 61 7 38335508; e-mail: shijie.zhou@csiro.au.

Liu, C., Zhou, S., Wang, Y.-G., and Hu, Z. Natural mortality estimation using tree-based ensemble learning models. – ICES Journal of Marine Science, 77: 1414–1426.

Received 21 August 2019; revised 4 March 2020; accepted 13 March 2020; advance access publication 5 June 2020.

Empirical studies are popular in estimating fish natural mortality rate (M). However, these empirical methods derive M from other life-history parameters and are often perceived as being less reliable than direct methods. To improve the predictive performance and reliability of empirical methods, we develop ensemble learning models, including bagging trees, random forests, and boosting trees, to predict M based on a dataset of 256 records of both Chondrichthyes and Osteichthyes. Three common life-history parameters are used as predictors: the maximum age and two growth parameters (growth coefficient and asymptotic length). In addition, taxonomic variable class is included to distinguish Chondrichthyes and Osteichthyes. Results indicate that tree-based ensemble learning models significantly improve the accuracy of M estimate, compared to the traditional statistical regression models and the basic regression tree model. Among ensemble learning models, boosting trees and random forests perform best on the training dataset, but the former performs a slightly better on the test dataset. We develop four boosting trees models for estimating M based on varying life-history parameters, and an R package is provided for interested readers to estimate M of their new species.

Keywords: empirical methods, ensemble learning methods, life-history parameters, natural mortality, regression tree, statistical learning

Introduction

Various empirical (indirect) methods have been developed to estimate natural mortality rate (M) of fish based on surrogate life-history parameters (Pauly, 1980; Hoening, 1983). The commonly used estimators can be divided into four groups according to the parameters used in the formula: (i) based on maximum age, t_{\max} (Bayliff, 1967; Hoening, 1983; Hewitt and Hoening, 2005), (ii) based on von Bertalanffy growth coefficient K (Beverton, 1963; Jensen, 2001), (iii) based on growth parameters K and L_{∞} and with or without water temperature T (Pauly, 1980; Roff, 1986;

Gulland, 1987), and (iv) based on both K and t_{\max} (Alverson and Carney, 1975; Zhang and Megrey, 2006). However, these empirical methods are often perceived as being less reliable than direct methods, such as mark-recapture (Brooks *et al.*, 1998; Hewitt *et al.*, 2007) and telemetry techniques (Hightower *et al.*, 2001; Heupel and Simpfendorfer, 2002), as the fundamental form of relationship between M and surrogate life-history parameters is generally unknown, as well as life-history parameters themselves are often measured with errors (Hamel, 2014; Rudd *et al.*, 2019).

For example, Kenchington (2014) reviewed 30 M estimators and found that none of them can provide accurate estimation for every species and none appears sufficiently precise for use in analytical stock assessments, while several perform so poorly as to have no practical utility.

Another concern is the application of empirical methods for estimating natural mortality in Chondrichthyes because a very small number of Chondrichthyes have been included in the dataset used to develop empirical estimators (Braccini *et al.*, 2017; Smart *et al.*, 2018; Harry *et al.*, 2019). Frisk *et al.* (2001) found that the link between M and life-history parameters for cartilaginous fish was significantly different from those of other taxonomic groups. However, there are very few studies on M estimation for Chondrichthyes. Most studies have focused on the Osteichthyes hitherto (Djabali *et al.*, 1993; Jensen, 2001; Griffiths and Harrod, 2007). For example, Then *et al.* (2015) compared and ranked the predictive abilities of the four major empirical estimation approaches and finally recommended two updated estimators: Hoenig_{nls} estimator ($M_{\text{est}} = 4.899t_{\text{max}}^{-0.916}$) and Pauly_{nls-T} estimator ($M_{\text{est}} = 8.87K^{0.73}L_{\infty}^{-0.33}$). More than 200 records were used in that study, but only four (<2%) are Chondrichthyes (all in order Carcharhiniformes). When we tested these two estimators recommended by Then *et al.* (2015) on a new dataset of 60 Chondrichthyes samples, we found that both estimators overestimated M for Chondrichthyes and Pauly_{nls-T}'s estimator, in particular, significantly overestimated M for almost all 60 Chondrichthyes.

Clearly, using estimators derived from Osteichthyes to predict M of Chondrichthyes can produce incorrect estimates and consequently lead to wrong stock assessment and fisheries management decisions.

Recently, the use of statistical learning methods, such as tree-based methods for classification and regression, is becoming more and more commonplace in the bio-medical field (Li and Wong, 2003; Tomar and Agarwal, 2013) and a myriad of other domains (Tso and Yau, 2007), including fishery research (Walsh and Kleiber, 2001; Soykan *et al.*, 2014; Zhou *et al.*, 2018). Tree-based regression models have various advantages, including effectively handling the quantitative and qualitative information simultaneously without pre-processing. In addition, these models do not need to specify the form of the predictors' relationship to the response (Max, 2008).

This study has two innovative features. First, we compiled a dataset containing 196 records of teleost fish and 60 records of cartilaginous fish from existing datasets and literature. Second, considering the difference of M between Chondrichthyes and Osteichthyes, we developed tree-based ensemble learning models, including bagging trees, random forests, and boosting trees using this dataset to estimate M of Chondrichthyes and Osteichthyes simultaneously and to improve its prediction accuracy. The accuracy of four tree-based models is compared to the traditional statistical regression models. The study demonstrates the advantages of regression tree and ensemble learning models in estimating natural mortality rates and suggests potential use in other areas of fish and fisheries research.

Material and methods

Data sources

We conducted a literature search and collected the necessary data from a variety of sources, including published research papers, reports, and grey documents. The key information we were interested in included directly estimated M and associated life-history

parameters on maximum age (t_{max}) and von Bertalanffy growth parameters (K and L_{∞}).

Some M estimators (such as Pauly, 1980) include water temperature as another predictor in addition to life-history parameters. However, Gislason *et al.* (2010) and Then *et al.* (2015) confirmed that the correlation between temperature and natural mortality was weak so they excluded water temperature in their M estimators. We adopted their approaches and focused on life-history parameters.

After carefully checking and reconfirming the source of M estimates, we adopted 196 of the 230 samples compiled by Then *et al.* (2015). Their data were composed of existing compilations of estimates of M (including Pauly, 1980; Hoening, 1983; Gislason *et al.*, 2010) as well as their own literature searches. In addition, from individual published literature, we collected additional 60 estimates of M for Chondrichthyes. We examined the original studies and their methods used to derive these M estimates. The data used in this study contains a total of 256 records from 2 classes (Chondrichthyes and Osteichthyes), 28 orders, 70 families, and 223 species (see Supplementary Table S1).

Our dataset included both commercial and non-commercial fish stocks. All the data were critically reviewed and compiled according to the following criteria:

- (i) We used only independently estimated M obtained from, for example directly estimated from tagging mark-recapture studies, telemetry studies or visual census (Grant *et al.*, 1979; Knip *et al.*, 2012), population dynamics models (Fletcher, 1995; Cortés and Parsons, 1996), and field observations (Hutchings and Griffiths, 2010). M estimated from previously published empirical relationships like Hoening (1983) are excluded.
- (ii) If M estimate is a new estimate made based on already published data that cannot be fully identified as direct estimates, for example 33 samples that are marked by * in Then *et al.* (2015) are excluded.
- (iii) Total mortality based on catch-at-length (Cortés and Parsons, 1996) or catch-at-age (Williams *et al.*, 2008) data is adopted if the data come from an unexploited or just lightly exploited stock.
- (iv) We examined and re-verified each of the original and directly estimated M , excluding some data with obvious errors, or extremely rare species, such as the miniature species of coral reef fish from the genus *Eviota* (Depczynski and Bellwood, 2006), which has a natural mortality rate as high as 50 (year^{-1}), while the natural mortality rate of all other fish is <8 (year^{-1}).
- (v) All parameters for the same species are derived from the same study if possible, or from the studies upon same species of the same location and timing of the study.
- (vi) Not all three life-history parameters (i.e. K , L_{∞} , and t_{max}) are available for all stocks. For example, there are nine records without growth parameters (K and L_{∞}) and three records without maximum age (t_{max}). However, these samples with one or two missing variables can still be used in tree-based models. Unlike traditional predictive models, tree-based models can specifically account for missing data by adjusting the information gain statistic since information gain is an important indicator of attribute selection in the process of decision tree construction. For example, when the predictor contains

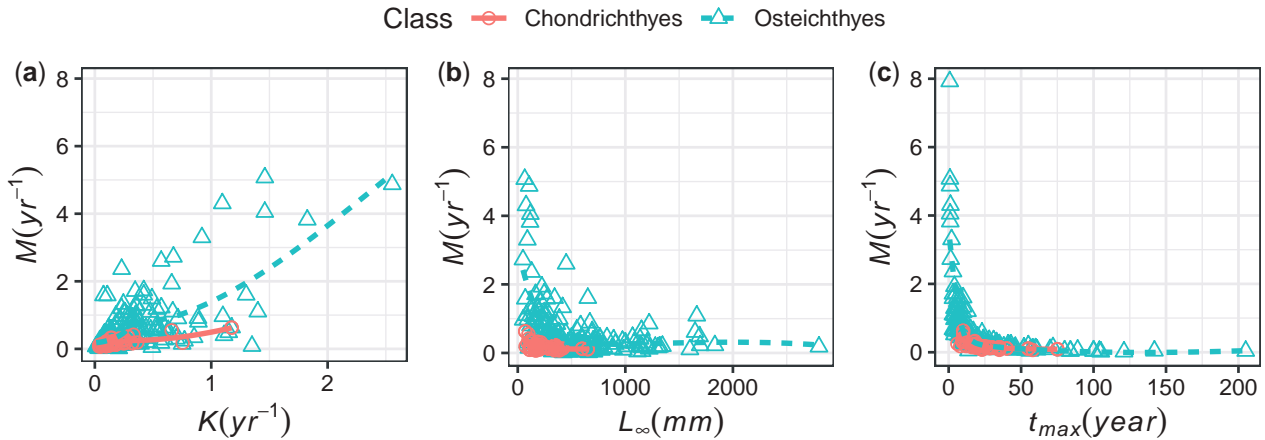


Figure 1. Scatter plot of M over three life-history parameters used as predictors.

missing value, the number of branches is increased by one; missing data are treated as an “extra” category or value of the predictor (Max and Kjell, 2013).

Among life-history parameters, t_{\max} , K , and L_{∞} are widely available and clearly correlate with M (Beverton, 1992; Kenchington, 2014; Then et al., 2015). However, the non-linear relationship between M and these life-history parameters may not be simply described by a mathematical equation. The M distribution of Osteichthyes and Chondrichthyes shows significantly different characteristics. In addition, it is not uncommon when multiple M correspond to a same predictor value in different studies (Figure 1). In these cases, M may vary widely across the values of predictors. For these reasons, partitioning methods, such as regression trees, may be able to isolate the predictors within the model and more effectively predict M .

It is well known that fish belong to a same taxonomic group possess similar trait (Zhou et al., 2012; Thorson et al., 2017). Hence, in addition to life-history parameters, we include taxonomy (class) as a categorical (qualitative) variable in the models, so the predictors used are t_{\max} (years), K (year^{-1}), L_{∞} (mm), and class.

Tree-based ensemble learning methods

Basic regression tree

Classification and Regression Trees algorithm is a binary tree, which can be used for both classification and regression problems according to the types of response variable. Tree-based regression methods involve segmenting the predictor space into several simple regions. Regression trees typically use the mean value of the training observations in the region to which it belongs to make a prediction for a given observation (Breiman et al., 1984). Tree-based regression methods are ideal for dealing with real-value prediction, such as M , for several reasons. First, they generate a set of conditions that are highly interpretable and are easy to implement. They can effectively handle many types of predictors (sparse, skewed, continuous, categorical, etc.) without the need to pre-process them (e.g. data transformation) (Strobl et al., 2009). In addition, these models do not require the user to specify the form of the predictors’ relationship to the response, for example linear, polynomial, or exponential relationship (Max and Kjell, 2013).

However, the basic regression tree may not be competitive with the best supervised learning approaches in terms of prediction accuracy. In this study, we use several ensemble learning models, i.e. bagging trees, random forests, and boosting trees, that combine many trees into one model to yield a single consensus prediction. Ensemble learning methods can often result in dramatic improvements in prediction accuracy, at the expense of some loss in interpretation (Dietterich, 2000).

Bagging trees

The basic idea of bagging trees is to generate B different bootstrapped (Johnson, 2001) training datasets and then train the regression tree on the i th bootstrapped training set to get prediction $\hat{f}^{*i}(x)$ and finally average all the predictions to obtain $\hat{f}_{\text{bag}}(x) = 1/B \sum_{i=1}^B \hat{f}^{*i}(x)$, hence increasing the prediction accuracy (Breiman, 1996). These trees are grown deep and are not pruned, so each individual tree has high variance, but low bias. Averaging these B trees reduces the variance. Bagging has been demonstrated to give impressive improvements in accuracy by combining hundreds or even thousands of trees into a single procedure. It is illustrated simply in Algorithm 1.

Algorithm 1 Bagging trees

```

1 for  $i = 1$  to  $B$  do
2   Generate a bootstrap sample of the original data
3   Train an unpruned tree model on this sample
4 end

5 Output the bagging model:  $\hat{f}_{\text{bag}}(x) = 1/B \sum_{i=1}^B \hat{f}^{*i}(x)$ 

```


Algorithm 2 Random forests

```

1 Select the number of trees to build,  $B$ 
2 for  $i = 1$  to  $B$  do
3   Generate a bootstrap sample of the original data
4   Train a tree model on this sample
5   for each split do
6     Randomly select  $k (< p)$  of the original predictors
7     Select the best predictor among the  $k$  predictors and partition the data
8   end
9   Use typical tree model stopping criteria to determine when a tree is complete (but do
   not prune)
10 end
11 Output the random forests model:  $\hat{f}_{rf}(x) = 1/B \sum_{i=1}^B \hat{f}^{*i}(x)$ 

```

Algorithm 3 Boosting trees

```

1 Set  $\hat{f}(x) = 0$  and  $r_i = y_i$  for all  $i$  in the training set.
2 Compute the average response,  $\bar{y}$ , and use this as the initial predicted value for each
sample
3 for  $i = 1$  to  $B$  repeat:
4   Fit a tree  $\hat{f}^{*i}(x)$  with  $d$  splits ( $d + 1$  terminal nodes) to the training data
5   Update  $\hat{f}(x)$  by adding in a shrunken version of the new tree:  $\hat{f}(x) \leftarrow \hat{f}(x) +$ 
 $\lambda \hat{f}^{*i}(x)$ 
6   Update the residuals,  $r_i \leftarrow r_i - \lambda \hat{f}^{*i}(x)$ 
7 end
8 Output the boosting trees:  $\hat{f}(x) = \sum_{b=1}^B \lambda \hat{f}^{*b}(x)$ 

```

Random forests

Breiman (2001) proposed random forests, which provide an improvement over bagging trees by way of a random small tweak that reduces correlation among trees. As in bagging, we built a

few decision trees on bootstrapped training samples. When building these decision trees, random forests algorithm randomly selects a subset of k predictors at each split from the total p

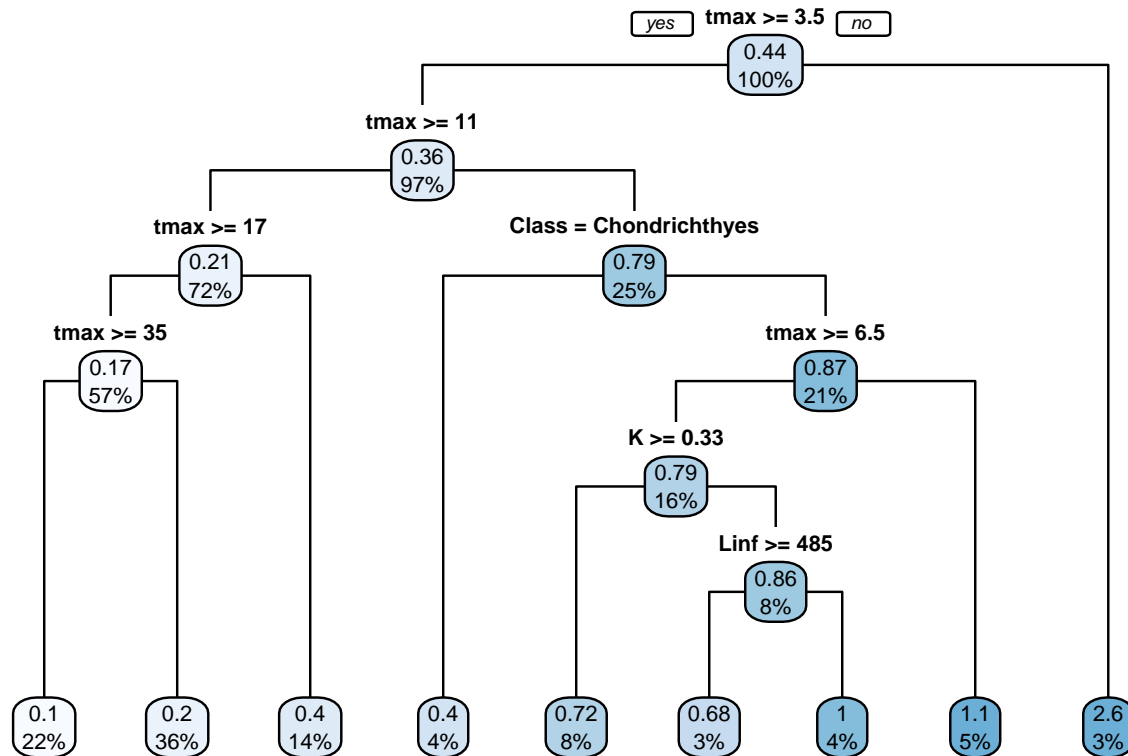


Figure 2. Basic regression tree model to predict M . For each split, the first line in the shaded box is M value and the second line is the percentage of the samples.

predictors. Therefore, tree correlation will necessarily be reduced. The random forests are illustrated in Algorithm 2.

Boosting trees

In bagging tree models, each tree is built on a bootstrap dataset, independent of the other trees. Boosting works in a similar way, except that the trees are grown sequentially: each tree is grown using information from previously grown trees (Schapire *et al.*, 1998). Boosting does not involve bootstrap sampling, instead each tree is fit on a modified version of the original dataset. The process of boosting trees is described in Algorithm 3. There are three important tuning parameters in boosting process: (i) the number of trees B . Unlike bagging and random forests, boosting can overfit if B is too large, although this overfitting tends to occur slowly if at all. We use cross-validation to select B . (ii) The shrinkage parameter λ , a small positive number. This controls the rate at which boosting learns. Typical values are 0.01 or 0.001, and the right choice can depend on the problem. Very small λ may require using a very large value of B to achieve good performance. (iii) The number d of splits in each tree, which controls the complexity of the boosted ensemble. Often $d = 1$ works well, in which case each tree is a stump, considering of a single split. More generally, d is the interaction depth and controls the interaction order of boosted model, since d splits can involve at most d variables (Max and Kjell, 2013).

Modelling process

We used a random sample of 75% of the data for training set and 25% of the data for holdout test set to create a series of models and evaluate them. First, a tenfold cross-validation with ten

repeats technique (Arlot and Celisse, 2010) was used to tune various models on the training set, since it gave the mean value of the k -division test results, which was more robust and more practical in bagging models. We then applied the models to the test set to evaluate the performance of the four models (i.e. basic regression tree, bagging trees, random forests, and boosting trees). Once the final model is selected, the model is used to predict M of new species. All results were produced using the R package “caret” (Max, 2008), which contains numerous tools for developing predictive models.

The tree-based methods require tuning several control parameters to achieve the best performance. For example, the number of trees needs to be determined in each ensemble learning model. For random forests, the number of predictors considered in each split also needs to be chosen. And for boosting trees, more parameters need to be determined, including shrinkage parameter that represents the learning rate of the models, interaction depth controlling the model complexity, and the minimum number of observations in trees’ terminal nodes.

The number of trees and the number of predictors considered in each split in random forests can be automatically determined by the tenfold cross-validations. In the boosting trees, for our dataset, a tuning parameter grid was constructed where interaction depth ranged from 1 to 10, number of trees ranged from 50 to 5000, shrinkage ranged from 0.001 to 0.01, and the minimum number of observations in trees’ terminal nodes ranged from 1 to 3. From these arrays, the best tuning parameter can be chosen by tenfold cross-validation with ten repeats technique as well.

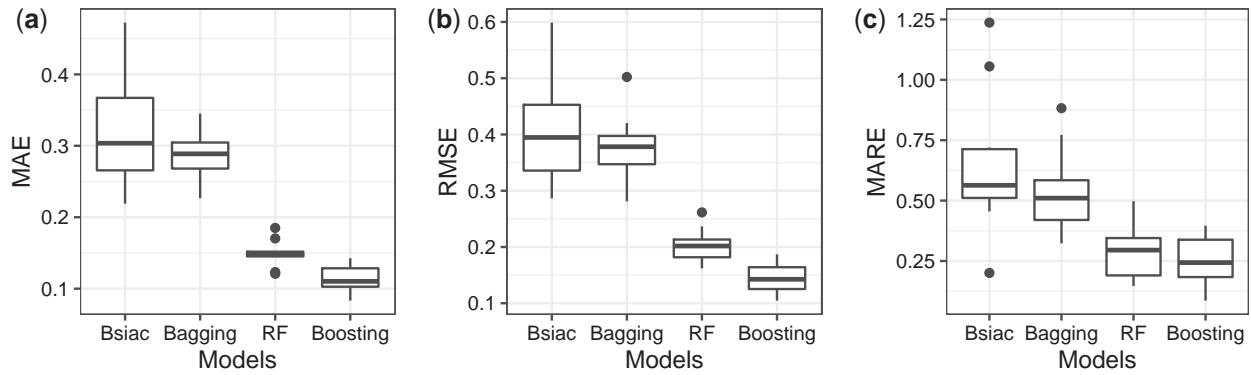


Figure 3. Comparison of four tree-based models by three performance metrics using training dataset. Basic, basic regression tree; bagging, bagging trees; RF, random forests; boosting, boosting trees.

Model performance assessment

We used three metrics to compare alternative models in this study: (i) mean absolute error ($MAE = 1/n \sum_{i=1}^n M_{obs,i} - M_{est,i}$), which measures the average absolute difference between observed and predicted outcomes; (ii) root mean squared error ($RMSE = \sqrt{1/n \sum_{i=1}^n (M_{obs,i} - M_{est,i})^2}$), which measures the average error performed by the model in predicting the outcome for an observation; and (iii) mean absolute relative error ($MARE = 1/n \sum_{i=1}^n (M_{obs,i} - M_{est,i})/M_{obs,i}$). We calculated the three metrics using both original and log-transformed values. However, only log-transformed results are presented in this study because with log-transformation the distribution of model residuals was closer to the normal distribution and the variance was more homogeneous.

Results

Interpretation of regression tree approaches

We first used a basic regression tree to illustrate how tree-based method works (Figure 2). It consists of a series of splitting rules, starting at the top of the tree. Overall, the tree segments M into nine regions (nine terminal nodes) of predictor space. The number in each terminal node is the mean of the response for the observations that fall into this space.

In this four-predictor model (t_{max} , K , L_{∞} , and class), the tree shown in Figure 2 indicates that t_{max} is the most important factor in determining M . Fish with larger t_{max} experience a lower M , and vice versa. When a species has a low t_{max} ($t_{max} < 3.5$), the taxonomic class plays little role in M prediction. When the contribution of class becomes significant, Chondrichthyes (class = 1) have lower M than Osteichthyes (class = 0). Therefore, an interaction effect between t_{max} and class occurs when the effect of explanatory variable t_{max} on the response variable M depends on the level of class. The interactions between other variables are presented in the same way. It is the hierarchical structure of a tree that guarantees the interactions between predictors to be automatically modelled (Elith *et al.*, 2008). In contrast to t_{max} , fish with higher values of K and L_{∞} experience a higher M as well. Figure 2 is likely an oversimplification of the true relationship between M and t_{max} , K , L_{∞} and class, though it is easy to interpret and has a nice graphical representation. More advanced ensemble learning methods described in this study follow the similar concept as in the basic regression tree.

Model fitting using training dataset

In machine learning, model training is to fit a model on training dataset. To determine which tree-based method performs best so it can be used to further develop predictive models for estimating M , we used all four predictors (t_{max} , K , L_{∞} , and class) to training the four models (i.e. basic regression tree, bagging trees, random forests, and boosting trees) on the training dataset. Figure 3 shows boxplots for the tenfold cross-validation with ten repeats resampling results across different models. The results obtained from three performance metrics (MAE, RMSE, and MARE) are consistent. The best-performing model is the boosting trees, followed by random forests. Bagging trees show modestly improved results relative to the basic tree but are clearly worse than the boosting trees and random forests. However, all these three ensemble learning methods outperform the basic regression tree model. The model rankings remain unchanged whether the data are log-transformed or not.

Model prediction using test dataset

The prediction results on test dataset (Table 1) are consistent with the cross-validation rankings in Figure 3. The best model is the boosting trees, followed by random forests and bagging trees. Boosting trees performs slightly better than random forests on test set, regardless of which performance metric is used. Visually, the residual ($M_{obs,i} - M_{est,i}$) diagnostic plots support the conclusion that the boosting trees performs best among four methods (Figure 4).

Model performance comparison using full dataset

After the best method is selected above, the prediction performance of boosting trees and the two estimators suggested by Then *et al.* (2015) are compared based on the newly compiled dataset of 256 samples. Here, we use the tenfold cross-validation repeat 20 times to estimate the test error of the boosting trees models. We build four models with varying predictors: BRT1 uses t_{max} and class; BRT2 uses K , L_{∞} , and class; BRT3 uses t_{max} , K , L_{∞} , and class; and BRT4 uses K and class. Again, the prediction RMSE (also referred to as cross-validation prediction errors), MAE and MARE are used as measurements of the model performance.

Clearly, boosting trees BRT1 performs better than the Hoenig_{nls} estimator when only t_{max} is used (Table 2). Boosting trees BRT2 also performs better than Pauly_{nls-T} estimator when

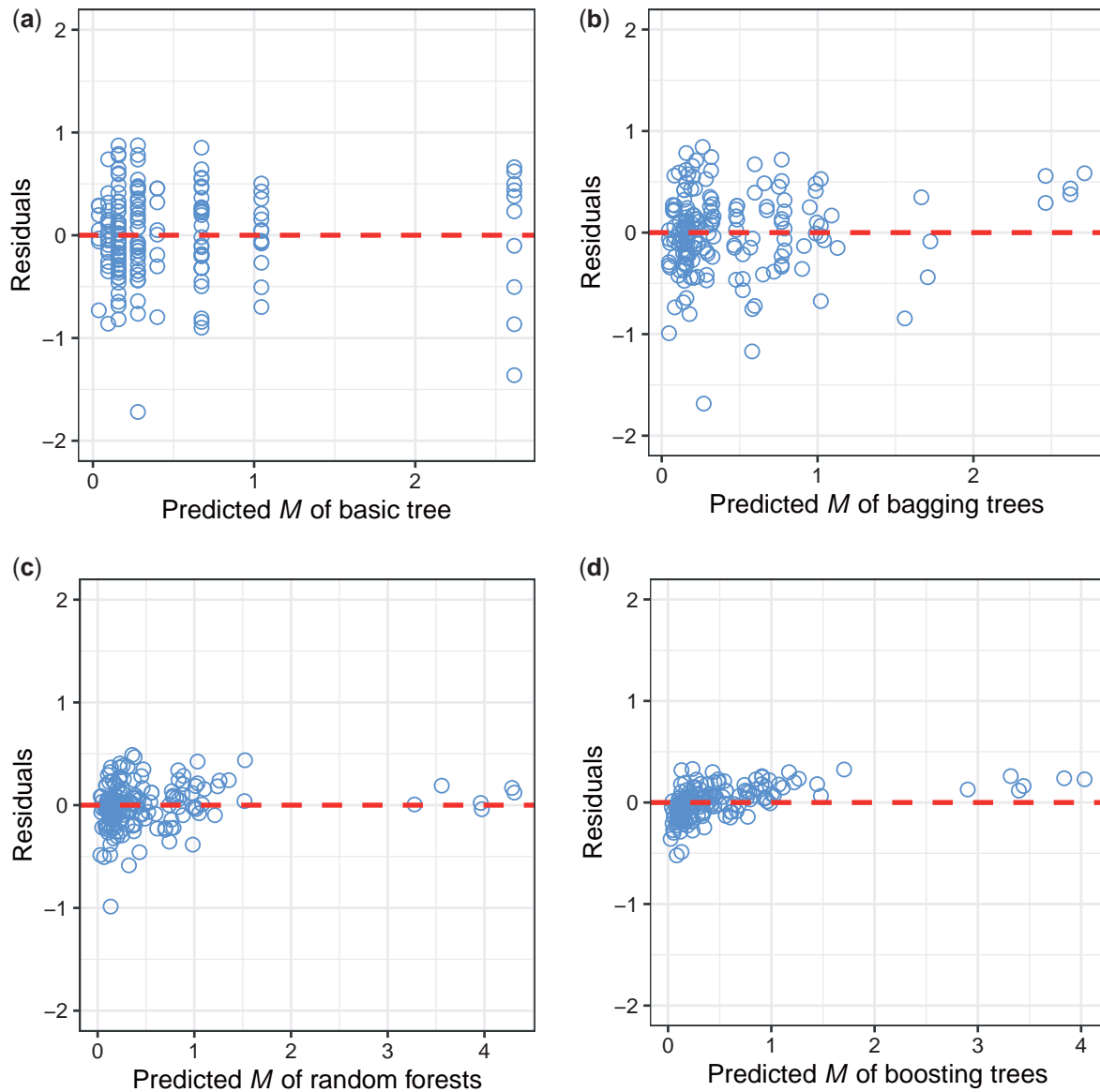


Figure 4. Residual diagnostic plots of four tree-based models on the test (validation) dataset.

Table 1. Performance of model prediction on test (validation) dataset.

Models	Model performance metrics		
	MAE	RMSE	MARE
Basic regression tree	0.32533	0.41155	0.65378
Bagging tree	0.28750	0.37698	0.53369
Random forests	0.14910	0.20346	0.28572
Boosting trees	0.11286	0.14407	0.24686

The models have four predictors (t_{\max} , K , L_{∞} , and class).

only K and L_{∞} are used in the model. Moreover, boosting trees produce unbiased estimates compared to $\text{Pauly}_{\text{nlS-T}}$ estimator that overestimates M when the value increases (Figure 5b). BRT3

that involves all four predictors significantly outperforms BRT1 against the three metrics (Table 2). Visually, BRT3 achieves the best results as well, with all the residuals within ± 0.5 (Figure 5c).

Although $\text{Hoenig}_{\text{nlS}}$ and $\text{Pauly}_{\text{nlS-T}}$ estimators were recommended by Then (2015), we noticed that the two-parameter K estimator was actually comparable to $\text{Pauly}_{\text{nlS-T}}$ estimator in the study of Then et al. (2015). Compared to the two-parameter K estimator in Then et al. (2015), our boosting trees model BRT4 with parameters K and class performs better against all performance metrics (Table 2). In addition, like the difference between BRT3 and BRT1, BRT2 model, which has one more parameter L_{∞} than BRT4, yields noticeable improvement over BRT4.

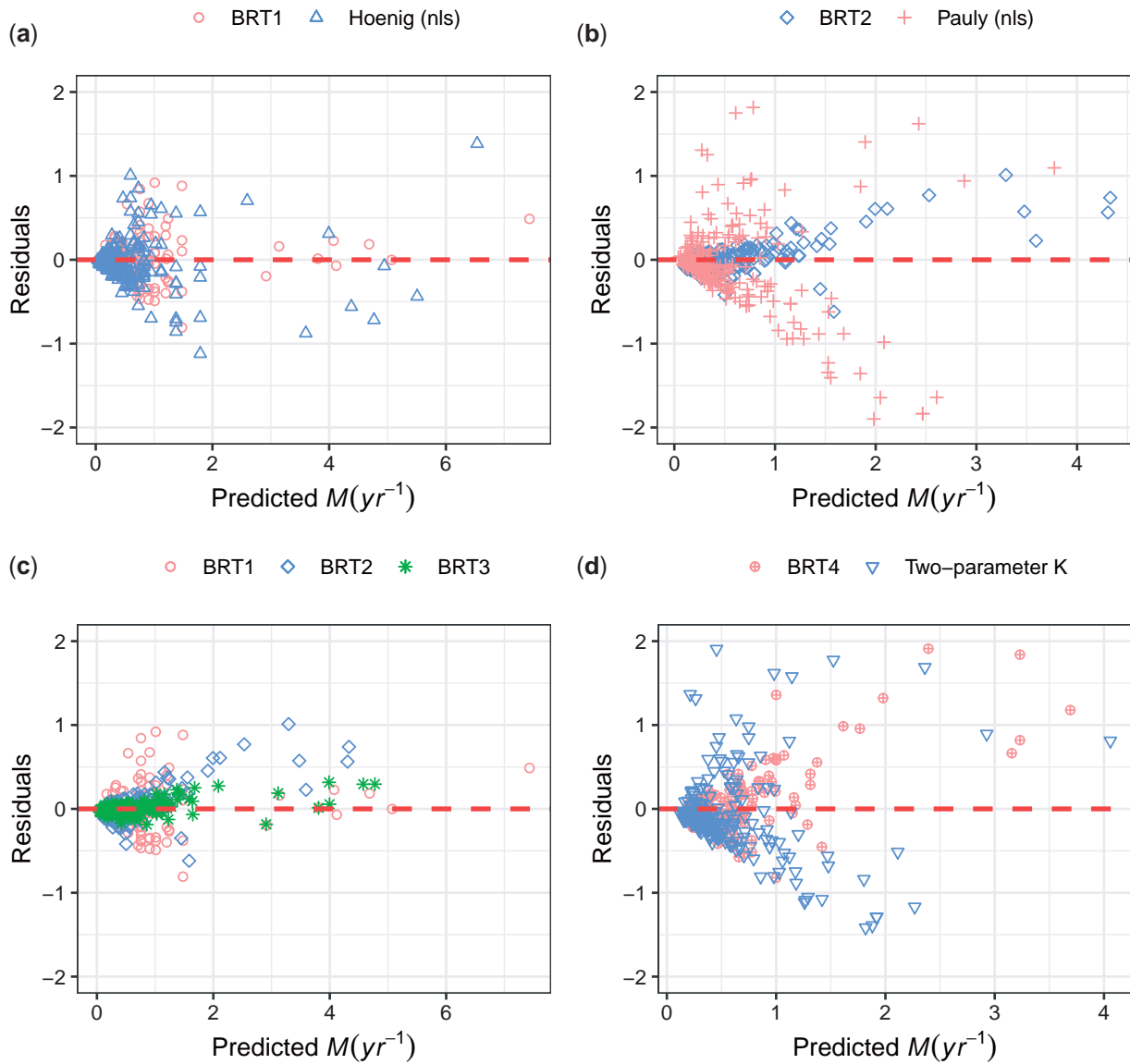


Figure 5. Comparison of predictive residuals between boosting trees models (BRT) and traditional regression models. BRT1 uses predictors t_{max} and class; BRT2 uses predictors K , L_{∞} , and class; BRT3 uses predictors t_{max} , K , L_{∞} , and class; and BRT4 uses predictors K and class.

Table 2. Model performance comparison using full dataset.

Model	Predictors	Model performance metrics		
		MAE	RMSE	MARE
BRT1	t_{max} , class	0.10665	0.18640	0.25020
Hoenig _{nls}	t_{max}	0.16942	0.25973	0.37875
BRT2	K , L_{∞} , class	0.12315	0.19638	0.23830
Pauly _{nls-T}	K , L_{∞}	0.35066	0.54028	0.59875
BRT3	t_{max} , K , L_{∞} , class	0.05790	0.08662	0.13779
BRT4	K , class	0.17321	0.27182	0.36590
Two-parameter K	K	0.35506	0.54681	0.67822

BRT1, boosting trees model with parameters t_{max} and class; BRT2, boosting trees model with parameters K , L_{∞} , and class; BRT3, boosting trees model with parameters t_{max} , K , L_{∞} , and class; BRT4, predictors K and class.

Table 3. The best boosting trees obtained by tenfold cross-validation with 20 repeats.

Model	Predictors	Tuning parameters			
		B	λ	d	N
BRT1	t_{max} , class	3900	0.001	45	1
BRT2	K , L_{∞} , class	1900	0.001	45	1
BRT3	t_{max} , K , L_{∞} , class	2700	0.001	45	1

BRT1, boosting trees model with parameters t_{max} and class; BRT2, boosting trees model with parameters K , L_{∞} , and class; B , number of trees; λ , shrinkage parameter; d , interaction depth; N , minimum number of observations in trees' terminal nodes.

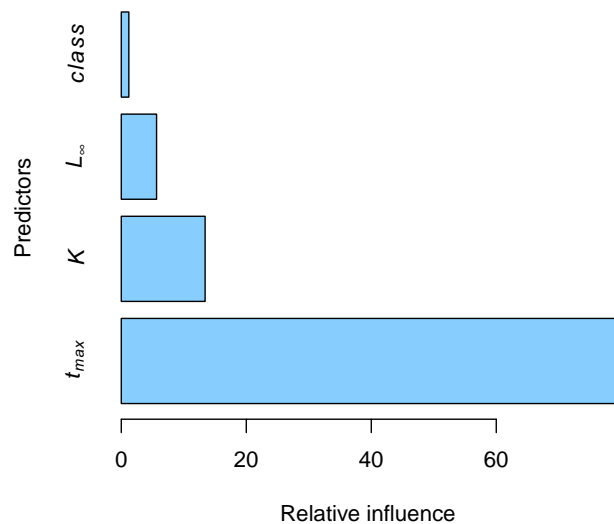


Figure 6. Relative importance of the four predictors in the boosting trees model.

Boosting trees—the best model

We further describe the detailed features of the boosting trees model that performs best on the fullest dataset. Based on the tenfold cross-validation with 20 repeats analyses, the optimal prediction can be achieved using the controlling parameters in Table 3.

Based on optimal values for the four controlling parameters, we can obtain the relative importance of each predictor in the model (Figure 6) by calculating the residual sum of squares [RSS = $\sum_{j=1}^J (M_{\text{obs},j} - M_{\text{est},j})^2$], where J represents the number of observations in the J th predictive space. In the case of boosting regression trees, we can record the total amount of RSS decreased due to splits over a given predictor, averaged over all B trees. A large value indicates an important predictor.

The most important variable in M prediction is t_{\max} , followed by K . Although the variable class only makes a small contribution compared to other predictors, its biological significance is clear: given identical life-history traits, Chondrichthyes' M is almost 0.02 year^{-1} lower than Osteichthyes (Figure 7).

Although the ensemble learning methods suffer from a lack of intuitive interpretability compared to a single regression tree and traditional linear regression models, the partial dependence plots (pdps) show how each predictor affects the model's predictions (Figure 7). Each panel illustrates the marginal effect of the selected variable on the response after integrating out the other variables. The volatility in the pdps indicates that the response variable M is very sensitive to changes in the predictors within this range, especially for the predictors K and L_{∞} . The flat segments mean that the predictors have stable effect on the response variable M within this range. For example, when t_{\max} is smaller than ~ 30 years, the value of M decreases rapidly as t_{\max} increases. However, when t_{\max} is greater than 30 years, M is much less dependent on t_{\max} (Figure 7a). The partial dependence of predictors K and L_{∞} shows obvious non-linear and piecewise characteristics. This is mainly because values of K and L_{∞} are not evenly distributed across their range and their relationships with M between Chondrichthyes and Osteichthyes are significantly different (Figure 1). For example, when K is larger than $1.3 \text{ (year}^{-1}\text{)}$, the

partial dependence effect of K on M jumps from about 0.5 to 0.9 (Figure 7b), which mainly because the partial dependence effect of variable K to M for Osteichthyes is much higher than that for Chondrichthyes and all Chondrichthyes have K smaller than $1.3 \text{ (year}^{-1}\text{)}$ (Figure 1a). The results of pdp appear to be consistency with Figure 1. When L_{∞} is smaller than 1000 mm, the predicted value of M gradually decreases as L_{∞} increases. When L_{∞} is between 1000 and 1500 mm, the partial dependence of M on L_{∞} increases with the increase of L_{∞} , which is mainly due to few data points and all of teleosts. The last panel in Figure 7 indicates that Chondrichthyes have a lower M than Osteichthyes even when other life-history parameters are identical.

Application of the boosting trees model to predict M of new species

We have developed an R package for our boosting trees models (see Supplementary Materials), and here, we provide examples for using the package to estimate M of new species. The examples include three flatfish species from Hamel (2014) and three shark species from Clarke et al. (2015): English sole (*Parophrys vetulus*), rex sole (*Glyptocephalus zachirus*), Petrale sole (*Eospetta jordani*), Blue shark (*Prionace glauca*), Shortfin mako shark (*Isurus paucus*), and Oceanic whitetip shark (*Carcharhinus longimanus*) (Table 4). Note that M values from literature may not be direct estimates but may also be based on life-history correlations.

Our R package *Mestimate* provides a function named "Mestimate". To estimate M for a new species that is not in our dataset, this function only requires one or more life-history parameters and an indicator of class (0 for Osteichthyes and 1 for Chondrichthyes) as inputs. This package provides three alternative boosting trees models according to the life-history parameters used for M estimation. Users can use one or all three models depending on the availability of life-history parameters. We recommend using the BRT1 (t_{\max} and class) model when only t_{\max} is available and BRT2 (K , L_{∞} , and class) model when only K and L_{∞} are available. When all life-history parameters are available, we recommend using the BRT3 (t_{\max} , K , L_{∞} , and class) model, which performs best in M estimation.

For the six species tested here, the results of BRT1 and BRT3 are comparable. However, like the results of Pauly_{nls-T} estimator, BRT2 produces a larger M for Osteichthyes than the other two boosting trees models (Table 4). Noticeable difference in some species estimated M exists between the boosting trees models and literature.

Discussion

Natural mortality rate is considered as an important but poorly quantified parameter in most mathematical models of fish stock dynamics (Vetter, 1988; Zhang and Megrey, 2006). In addition, existing empirical M estimators are almost always based on combined data for Osteichthyes and Chondrichthyes, ignoring the difference between these two groups of fish. In this study, we compile a new dataset containing 60 samples of Chondrichthyes and 196 samples of Osteichthyes. Our study demonstrates that tree-based regression methods can effectively estimate natural mortality rate of two classes of fish at the same time without the need to use dummy variables like traditional linear regression models do. More importantly, the tree-based ensemble learning models can significantly improve prediction accuracy compared to the traditional regression estimators suggested by the most

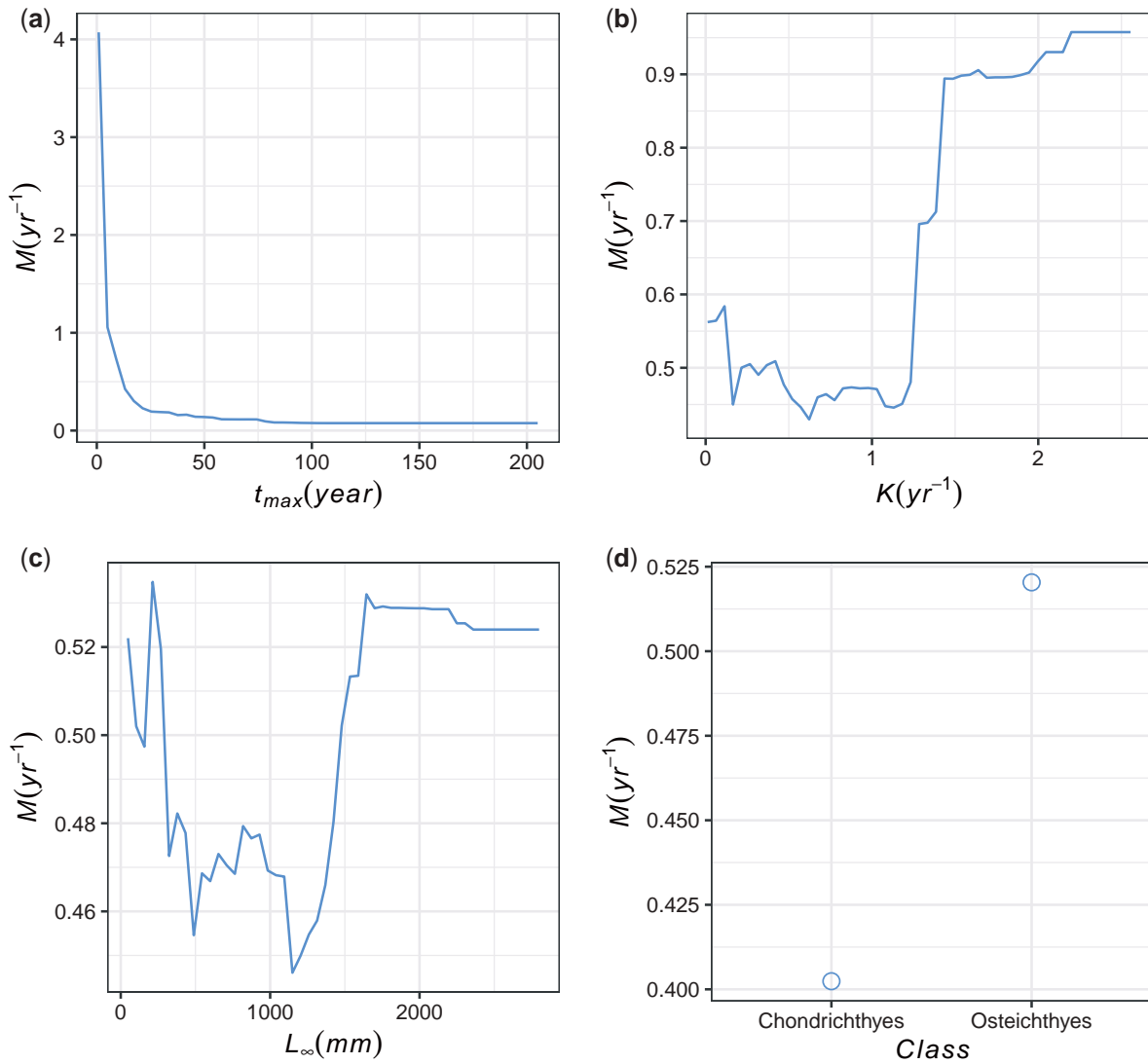


Figure 7. Partial dependence of predicted natural mortality on four predictors.

Table 4. Comparison of estimated M for six new species not used in model building.

Species	Predictors				M in literature M_{lit}	Predicted M by estimators in Then (2015)		Predicted M in this paper		
	t_{max}	K	L_{∞}	Class		Hoenig _{nls}	Pauly _{nls-T}	BRT1	BRT2	BRT3
PV	23	0.36	40.56	0	0.307	0.277	0.580	0.276	0.569	0.237
GZ	29	0.39	41.82	0	0.261	0.224	0.609	0.183	0.563	0.153
EJ	32	0.16	54.31	0	0.177	0.205	0.291	0.172	0.203	0.196
PG	15.5	0.142	327.4	1	0.273	0.398	0.316	0.194	0.238	0.242
IO	30.5	0.098	248.9	1	0.132	0.214	0.264	0.138	0.155	0.126
CL	21	0.095	262.6	1	0.180	0.301	0.253	0.156	0.153	0.138

PV, *Parophrys vetulus*; GZ, *Glyptocephalus zachirus*; EJ, *Eopsetta jordani*; PG, *Prionace glauca*; IO, *Isurus oxyrinchus*; CL, *Carcharhinus longimanus*. M_{lit} , M value from literature; BRT1, boosting trees model with parameters t_{max} and class; BRT2, boosting trees model with parameters K , L_{∞} , and class; BRT3, boosting trees model with parameters t_{max} , K , L_{∞} , and class.

comprehensive study (Then et al., 2015). Among the ensemble learning models, the boosting trees and random forests are the two best models, with the boosting trees performing a slightly better than random forests.

Although using various surrogate life-history parameters to estimate natural mortality rate has received extensive discussion in the literature (Jensen, 2001; Griffiths and Harrod, 2007; Mangel, 2017), to our knowledge this is the first study to estimate M using

ensemble learning technique. The major difference between our approach and the traditional regression models is that tree-based regression models do not need to specify a mathematical formula between the predictors and the response variable. The complex biological, physiological, and ecological processes often render a simple mathematical model problematic. Traditional regression also requires an assumption that there is no strong collinearity between the independent variables (Jensen, 2001). The boosting trees models generate a suit of different modified version of the original data set to create hundreds of models to reduce prediction bias and variance. In contrast to traditional regression, tree-based models do not require an assumption of no collinearity between predictors as the construction of each tree is based on a greedy algorithm that follows the problem-solving heuristic of making the locally optimal choice at each stage with the intent of finding a global optimum, so redundant features are not added to the model (Tomaschek et al., 2018). Therefore, we can use boosting trees model containing all available parameters to get the best prediction of M .

We only include taxonomic variable class in the ensemble learning models, noting that it is possible to involve finer level of taxonomy, such as order and family. We preliminarily explored a boosting trees model that includes class, order, and family as predictors. Because there are only 256 samples in our dataset, involving 28 orders and 70 families, the number of samples in each order and family is relatively small. Therefore, the results from these models have limited reference value and they are not reported in this study.

Our analysis supports the finding that t_{\max} -based estimator performs the best among all estimators evaluated (Then et al. 2015). Our analysis also demonstrates that asymptotic length only makes a minor contribution to model prediction. Therefore, when parameter L_{∞} is not available, the boosting trees model based on growth coefficient K and class can yield as good result as the model with all three predictors (K , L_{∞} , and class).

Although the tree-based ensemble learning models developed in this study can predict M reasonably well, these models only produce a mean value of the observations in terminal nodes of the regression trees (Breiman et al., 1984). Therefore, combining the traditional linear regression models and regression trees to create a hybrid model that produces a specific value of the M by a linear regression model in terminal nodes instead of the mean value of the observations may be a good way to produce better predictions and leads to better interpretability. This idea can be explored in the future studies. When boosting trees models are used for prediction, such as predicting M for a new species, the models currently could only give a point prediction of the response value but not the uncertainty around the point estimate. Instead, uncertainty estimated by cross-validation from model testing is usually adopted for the predicted M . Therefore, the RMSE shown in Table 2 represents the prediction error, which can be used as a proxy for the standard deviation.

Computing time may become a concern for boosting trees models when the dataset is large. Boosting trees models are grown sequentially: each tree is grown using the information from previously grown trees so it takes more time than other ensemble learning models, such as bagging trees and random forests.

Statistical modelling, including ensemble learning methods, can only be as good as the original data used to train the models. The data used in this study, whether from existing dataset or collected individually from literature, may contain high uncertainty.

Life-history parameters cannot be accurately measured. An examination of the life-history parameters compiled for Pacific sharks reveals high variability across studies (Zhou et al., 2019). In particular, maximum age may have been underestimated for many stocks because this parameter is either the observed or estimated maximum age from a population that has been fished for many years so very old fish rarely exist in the population. Sample sizes may also be inadequate (since the old fish are more likely to be included with larger sample sizes), and fishing may have selected smaller, younger fish, either through gear selectivity or because of fishing in areas where older fish are not present (Zhou et al., 2019). In teleost, t_{\max} is typically obtained from hard body parts such as scales or otoliths. Irregular early growth patterns or structural resorption near the primordium can lead to ageing error (Campana, 2001; Kolody et al., 2016). It is even more difficult to age Chondrichthyes as these cartilaginous fishes lack the large, calcareous otoliths (Francis et al., 2007). Usually, vertebrae are used for studies of Chondrichthyes growth and age. Recent studies show that the common method of ageing sharks and rays, counting growth zones on calcified structures, can substantially underestimate true age (Francis et al., 2007; Harry et al., 2019). If t_{\max} is severely underestimated in Chondrichthyes but not in Osteichthyes, the different result between the two classes as shown in Figure 7 may be spurious (i.e. smaller M for Chondrichthyes is due to their much large lifespan). The dependent variable M measured using direct or “information-intensive” estimation approach (Kenchington, 2014; Then et al., 2015) is also uncertain. Furthermore, all empirical approaches treat M as a constant for a particular species or stock, but natural mortality is rarely time-invariant (Vetter, 1988; Johnson et al., 2015). Therefore, ensemble learning models should be updated when new data become available. In addition, it is also possible to combine the Bayesian measurement error models with tree-based methods to incorporate measurement error in both dependent and independent variables in the future work.

Supplementary data

Supplementary material is available at the ICESJMS online version of the manuscript.

Acknowledgements

We thank three anonymous reviewers and Editor Dr. Poos for their constructive comments and suggestions that help us to improve the quality of the study.

Funding

This research was supported by the Australian Research Council Discovery Project (DP160104292) and the ARC Centre of Excellence for Mathematical and Statistical Frontiers (ACEMS), Australia.

References

- Alverson, D. L., and Carney, M. J. 1975. A graphic review of the growth and decay of population cohorts. *ICES Journal of Marine Science*, 36: 133–143.
- Arlot, S., and Celisse, A. 2010. A survey of cross-validation procedures for model selection. *Statistics Surveys*, 4: 40–79.
- Bayliff, W. H. 1967. Growth, mortality, and exploitation of the Engraulidae, with special reference to the anchoveta, *Cetengraulis mysticetus*, and the colorado, *Anchoa naso*, in the Eastern Pacific

- Ocean. Inter-American Tropical Tuna Commission Bulletin, 12: 365–432.
- Beverton, R. J. H. 1963. Maturation, growth and mortality of clupeid and engraulid stocks in relation to fishing. *Rapports et Procès-Verbaux des Réunions du Conseil Permanent International Pour l'Exploration de la Mer*, 154: 44–67.
- Beverton, R. J. H. 1992. Patterns of reproductive strategy parameters in some marine teleost fishes. *Journal of Fish Biology*, 41: 137–160.
- Braccini, M., Taylor, S., Bruce, B., and McAuley, R. 2017. Modelling the population trajectory of West Australian white sharks. *Ecological Modelling*, 360: 363–377.
- Breiman, L. 1996. Bagging predictors. *Machine Learning*, 24: 123–140.
- Breiman, L. 2001. Random forests. *Machine Learning*, 45: 5–32.
- Breiman, L., Friedman, H. J., Olshen, R. A., and Stone, C. J. 1984. *Classification and Regression Trees*. Wadsworth, Belmont, California.
- Brooks, E., Pollock, K., and Hoenig, J. S. H. W. 1998. Estimation of fishing and natural mortality from tagging studies on fisheries with two user groups. *Canadian Journal of Fisheries and Aquatic Sciences*, 55: 2001–2010.
- Campana, S. 2001. Accuracy, precision and quality control in age determination, including a review of the use and abuse of age validation methods. *Journal of Fish Biology*, 59: 197–242.
- Cortés, E., and Parsons, G. R. 1996. Comparative demography of two populations of the bonnethead shark (*Sphyrna tiburo*). *Canadian Journal of Fisheries and Aquatic Sciences*, 53: 709–718.
- Depczynski, M., and Bellwood, D. 2006. Extremes, plasticity, and invariance in vertebrate life history traits: insights from coral reef fishes. *Ecology*, 87: 3119–3127.
- Dietterich, T. G. 2000. Ensemble methods in machine learning. *In Multiple Classifier Systems*, Springer, Berlin. pp. 1–15.
- Djabali, F., Mehailia, A., Koudil, M., and Brahmi, B. 1993. Empirical equations for the estimation of natural mortality in Mediterranean teleosts. *Naga*, 16: 35–37.
- Elith, J., Leathwick, J. R., and Hastie, T. 2008. A working guide to boosted regression trees. *Journal of Animal Ecology*, 77: 802–813.
- Fletcher, W. 1995. Application of the otolith weight-age relationship for the pilchard, *Sardinops sagax neopilchardus*. *Canadian Journal of Fisheries and Aquatic Sciences*, 52: 657–664.
- Francis, M. P., Campana, S. E., and Jones, C. M. 2007. Age under-estimation in New Zealand porbeagle sharks (*Lamna nasus*): is there an upper limit to ages that can be determined from shark vertebrae? *Marine and Freshwater Research*, 58: 10–23.
- Frisk, M., Miller, T., and Fogarty, M. J. 2001. Estimation and analysis of biological parameters in elasmobranch fishes: a comparative life history study. *Canadian Journal of Fisheries and Aquatic Sciences*, 58: 969–981.
- Gislason, H., Niels, D., Rice, J., and G Pope, J. 2010. Size, growth, temperature and the natural mortality of marine fish. *Fish and Fisheries*, 11: 149–158.
- Grant, C. J., Sandland, R. L., and Olsen, A. M. 1979. Estimation of growth, mortality and yield per recruit of the Australian School Shark, *Galeorhinus Australis* (Macleay), from tag recoveries. *Marine and Freshwater Research*, 30: 625–637.
- Griffiths, D., and Harrod, C. 2007. Natural mortality, growth parameters, and environmental temperature in fishes revisited. *Canadian Journal of Fisheries and Aquatic Sciences*, 64: 249–255.
- Gulland, J. A. 1987. Natural mortality and size. *Marine Ecology Progress Series*, 39: 197–199.
- Hamel, O. S. 2014. A method for calculating a meta-analytical prior for the natural mortality rate using multiple life history correlates. *ICES Journal of Marine Science*, 72: 62–69.
- Harry, A. V., Butcher, P. A., Macbeth, W. G., Morgan, J. A. T., Taylor, S. M., and Geraghty, P. T. 2019. Life history of the common blacktip shark, *Carcharhinus limbatus*, from central eastern Australia and comparative demography of a cryptic shark complex. *Marine and Freshwater Research*, 70: 834–848.
- Heupel, M. R., and Simpfendorfer, C. A. 2002. Estimation of mortality of juvenile blacktip sharks, *Carcharhinus limbatus*, within a nursery area using telemetry data. *Canadian Journal of Fisheries and Aquatic Sciences*, 59: 624–632.
- Hewitt, D., and Hoenig, J. 2005. Comparison of two approaches for estimating natural mortality based on longevity. *Fishery Bulletin*, 103: 433–437.
- Hewitt, D. A., Lambert, D. M., Hoenig, J. M., Lipcius, R. N., Bunnell, D. B., and Miller, T. J. 2007. Direct and indirect estimates of natural mortality for Chesapeake Bay blue crab. *Transactions of the American Fisheries Society*, 136: 1030–1040.
- Hightower, J. E., Jackson, J. R., and Pollock, K. H. 2001. Use of telemetry methods to estimate natural and fishing mortality of striped bass in Lake Gaston, North Carolina. *Transactions of the American Fisheries Society*, 130: 557–567.
- Hoening, J. M. 1983. Empirical use of longevity data to estimate mortality rates. *Fishery Bulletin*, 81: 898–903.
- Hutchings, K., and Griffiths, M. H. 2010. Life-history strategies of *Umbrina robinsoni* (Sciaenidae) in warm-temperate and subtropical South African marine reserves. *African Journal of Marine Science*, 32: 37–53.
- Jensen, A. L. 2001. Comparison of theoretical derivations, simple linear regressions, multiple linear regression and principal components for analysis of fish mortality, growth and environmental temperature data. *Environmetrics*, 12: 591–598.
- Johnson, K. F., Monnahan, C. C., McGilliard, C. R., Vert-pre, K. A., Anderson, S. C., Cunningham, C. J., Hurtado-Ferro, F. *et al.* 2015. Time-varying natural mortality in fisheries stock assessment models: identifying a default approach. *ICES Journal of Marine Science*, 72: 137–150.
- Johnson, R. W. 2001. An introduction to the bootstrap. *Teaching Statistics*, 23: 49–54.
- Kenchington, T. J. 2014. Natural mortality estimators for information-limited fisheries. *Fish and Fisheries*, 15: 533–562.
- Knip, D. M., Heupel, M. R., and Simpfendorfer, C. A. 2012. Mortality rates for two shark species occupying a shared coastal environment. *Fisheries Research*, 125: 184–189.
- Kolody, D. S., Eveson, J. P., and Hillary, R. M. 2016. Modelling growth in tuna RFMO stock assessments: current approaches and challenges. *Fisheries Research*, 180: 177–193.
- Li, J., and Wong, L. 2003. Using rules to analyse bio-medical data: a comparison between C4.5 and PCL. *In Advances in Web-Age Information Management*, Springer, Berlin. pp. 254–265.
- Mangel, M. 2017. The inverse life-history problem, size-dependent mortality and two extensions of results of Holt and Beverton. *Fish and Fisheries*, 18: 1192–1200.
- Max, K. 2008. Building Predictive Models in R Using the caret Package. *Journal of Statistical Software*, 28: 1–26.
- Max, K., and Kjell, J. 2013. Regression trees and rule-based models. *In Applied Predictive Modeling*, Springer, New York. pp. 173–220.
- Pauly, D. 1980. On the interrelationships between natural mortality, growth parameters, and mean environmental temperature in 175 fish stocks. *ICES Journal of Marine Science*, 39: 175–192.
- Roff, D. A. 1986. Predicting body size with life history models. *BioScience*, 36: 316–323.
- Rudd, M. B., Thorson, J. T., Sagarese, S. R., and Kuparinen, A. 2019. Ensemble models for data-poor assessment: accounting for uncertainty in life-history information. *ICES Journal of Marine Science*, 76: 870–883.
- Schapire, R. E., Freund, Y., Bartlett, P., and Lee, W. S. 1998. Boosting the margin: a new explanation for the effectiveness of voting methods. *The Annals of Statistics*, 26: 1651–1686.



- Clarke, S., Coelho, R., Francis, M., Kai, M., Kohin, S., Liu, K.-M., Simpfendorfer, C. *et al.* 2015. Report of the Pacific Shark Life History Expert Panel Workshop, 28-30 April 2015. Western and Central Pacific Fisheries Commission, Scientific committee Eleventh Regular Session, Pohnpei, Federated States of Micronesia.
- Smart, J. J., Punt, A. E., Espinoza, M., White, W. T., and Simpfendorfer, C. A. 2018. Refining mortality estimates in shark demographic analyses: a Bayesian inverse matrix approach. *Ecological Applications*, 28: 1520–1533.
- Soykan, C. U., Eguchi, T., Kohin, S., and Dewar, H. 2014. Prediction of fishing effort distributions using boosted regression trees. *Ecological Applications*, 24: 71–83.
- Strobl, C., Malley, J., and Tutz, G. 2009. An introduction to recursive partitioning: rationale, application, and characteristics of classification and regression trees, bagging, and random forests. *Psychological Methods*, 14: 323–348.
- Then, A. Y., Hoenig, J. M., Hall, N. G., and Hewitt, D. A. 2015. Evaluating the predictive performance of empirical estimators of natural mortality rate using information on over 200 fish species. *ICES Journal of Marine Science*, 72: 82–92.
- Thorson, J. T., Munch, S. B., Cope, J. M., and Gao, J. 2017. Predicting life history parameters for all fishes worldwide. *Ecological Applications*, 27: 2262–2276.
- Tomar, D., and Agarwal, S. 2013. A survey on Data Mining approaches for Healthcare. *International Journal of Bioscience and Biotechnology*, 5: 241–266.
- Tomaschek, F., Hendrix, P., and Baayen, R. H. 2018. Strategies for addressing collinearity in multivariate linguistic data. *Journal of Phonetics*, 71: 249–267.
- Tso, G. K. F., and Yau, K. K. W. 2007. Predicting electricity energy consumption: a comparison of regression analysis, decision tree and neural networks. *Energy*, 32: 1761–1768.
- Vetter, E. F. 1988. Estimation of natural mortality in fish stocks: a review. *Fishery Bulletin*, 86: 25–43.
- Walsh, W. A., and Kleiber, P. 2001. Generalized additive model and regression tree analyses of blue shark (*Prionace glauca*) catch rates by the Hawaii-based commercial longline fishery. *Fisheries Research*, 53: 115–131.
- Williams, A. J., Currey, L. M., Begg, G. A., Murchie, C. D., and Ballagh, A. C. 2008. Population biology of coral trout species in eastern Torres Strait: implications for fishery management. *Continental Shelf Research*, 28: 2129–2142.
- Zhang, C.-I., and Megrey, B. A. 2006. A revised Alverson and Carney model for estimating the instantaneous rate of natural mortality. *Transactions of the American Fisheries Society*, 135: 620–633.
- Zhou, S., Deng, A. R., Hoyle, S., and Dunn, M. 2019. Identifying appropriate reference points for elasmobranchs within the WCPFC. WCPFC-SC14-2018/MI-WP-07. Report to the Western and Central Pacific Fisheries Commission Scientific CommiJee. Fourteenth Regular Session, 8–16 August 2018, Busan, Korea.
- Zhou, S., Punt, A. E., Smith, A. D. M., Ye, Y., Haddon, M., Dichmont, C. M., and Smith, D. C. 2018. An optimized catch-only assessment method for data poor fisheries. *ICES Journal of Marine Science*, 75: 964–976.
- Zhou, S., Yin, S., Thorson, J. T., Smith, A. D., Fuller, M., and Walters, C. J. 2012. Linking fishing mortality reference points to life history traits: an empirical study. *Canadian Journal of Fisheries and Aquatic Sciences*, 69: 1292–1301.

Handling editor: Jan Jaap Poos

Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Original Article

Developing a microscopic image dataset in support of intelligent phytoplankton detection using deep learning

Qiong Li ¹, Xin Sun ^{1*}, Junyu Dong¹, Shuqun Song², Tongtong Zhang¹, Dan Liu¹, Han Zhang¹, and Shuai Han¹

¹College of Information Science and Engineering, Ocean University of China, 238 Songling Road, Qingdao 266100, China

²CAS Key Laboratory of Marine Ecology and Environmental Sciences, Institute of Oceanology, Chinese Academy of Sciences, 7 Nanhai Road, Qingdao 266071, China

*Corresponding author: tel: + 86 532 6678 1729; e-mail: sunxin@ouc.edu.cn.

Li, Q., Sun, X., Dong, J., Song, S., Zhang, T., Liu, D., Zhang, H., and Han, S. Developing a microscopic image dataset in support of intelligent phytoplankton detection using deep learning. – ICES Journal of Marine Science, 77: 1427–1439.

Received 4 May 2019; revised 16 August 2019; accepted 19 August 2019; advance access publication 20 September 2019.

Phytoplankton plays an important role in marine ecological environment and aquaculture. However, the recognition and detection of phytoplankton rely on manual operations. As the foundation of achieving intelligence and releasing human labour, a phytoplankton microscopic image dataset PMID2019 for phytoplankton automated detection is presented. The PMID2019 dataset contains 10 819 phytoplankton microscopic images of 24 different categories. We leverage microscopes to collect images of phytoplankton in the laboratory environment. Each object in the images is manually labelled with a bounding box and category of ground-truth. In addition, living cells move quickly making it difficult to capture images of them. In order to generalize the dataset for *in situ* applications, we further utilize Cycle-GAN to achieve the domain migration between dead and living cell samples. We built a synthetic dataset to generate the corresponding living cell samples from the original dead ones. The PMID2019 dataset will not only benefit the development of phytoplankton microscopic vision technology in the future, but also can be widely used to assess the performance of the state-of-the-art object detection algorithms for phytoplankton recognition. Finally, we illustrate the performances of some state-of-the-art object detection algorithms, which may provide new ideas for monitoring marine ecosystems.

Keywords: deep learning, microscopic image, object detection, phytoplankton dataset

Introduction

Marine phytoplankton is the foundation of marine ecosystems (Charlson *et al.*, 1987). It is an ecological concept that refers to tiny plants floating in water. As one most important primary producer in the ocean and the global ecological environment, phytoplankton activates the marine food chain. Consequently, some marine shellfish can accumulate poisonous phytoplankton. Phytoplankton also participates in the biogeochemical cycle of biogenic elements such as carbon, nitrogen, and phosphorus. In addition to its important ecological significance, it also plays a vital role in aquaculture. Some harmful ecological phenomena, such as red tides and canola

in coastal areas of China, are all caused by marine phytoplankton, which can directly lead to the death of numerous aquatic organisms because of the lack of oxygen.

Recently, the research of phytoplankton community structures mainly relies on scientific researchers to manually identify and count through microscopes, most of which belongs to the non-*in situ* category of observation methods. These traditional methods are time-consuming, labour-intensive, and require a high level of professional knowledge. In the past few years, automated identification of phytoplankton has drawn lots of attention. However, researchers only focused on the individual parts of automated recognition, such as

image processing and image segmentation. Besides, only common image features have been used for the automated identification algorithms of phytoplankton and FlowCAM (Poulton, 2016) is one of the more advanced technologies. It is an intelligent system for the automated analysis and sorting of phytoplankton cells. It can simultaneously measure multiple parameters of each cell and classify them according to their special characteristics. Although FlowCAM can directly measure seawater samples rapidly, qualitatively, and quantitatively, only a few traditional technologies are implemented such as clustering and image processing. Therefore, a huge improvement is urgent, especially with state-of-the-art artificial intelligence methods.

Deep learning (Deng and Yu, 2014; Lecun et al., 2015; Schmidhuber, 2015) is about learning multiple levels of representation that helps to make sense of data such as images (Ren et al., 2015; He et al., 2016), audio (Deng, 2014; Noda et al., 2015; Badjatiya et al., 2017; Fayek et al., 2017; Yu et al., 2017), and text (Lopez and Kalita, 2017; Young et al., 2018). It has made great achievements in many fields, especially in computer vision (Rawat and Wang, 2017; Sun et al., 2019), audio recognition

(Yu et al., 2017), and natural language processing (Watanabe et al., 2018; Young et al., 2018). One key of the success of deep learning is large training dataset. However, tedious and inefficient data annotating progress present an obstacle to further development of deep learning models. The model will not extract effective and distinctive features and lead to overfitting problems without enough train data (Srivastava et al., 2014). As a consequence, whether classification, recognition, or detection task, the use of deep learning methods requires a large amount of labelled data for model training. Especially in the process of microscopic observation of phytoplankton, where there is more than one object in a view. Researchers have to recognize and count each object in the given view, which is equivalent to object detection in the field of computer vision. In order to introduce the advanced deep learning approaches into the field of phytoplankton recognition, a phytoplankton dataset for deep learning is necessary.

It is difficult to train a deep learning model with strong generalization performance on the existing phytoplankton datasets. The existing datasets have several difficulties for training the deep learning methods as shown in Figure 1. First, the resolution of



Figure 1. PMID2019 is the first high resolution dataset for phytoplankton detection, compared with the existing datasets as shown above, ZooScan, WHOI, ASLO, and Kaggle-Plankton. The images from the various datasets are scaled to show the relative resolutions of the datasets.

phytoplankton images is too low to extract distinct features in detail. Second, the phytoplankton images are mostly grey-scale images. Third, the instance-level annotations are urgently needed, while current image datasets only have image-level annotations. Therefore, it is difficult to introduce the deep learning methods of detection with these existing datasets.

In this work, we focus on constructing a new phytoplankton dataset of high-resolution colourful images along with instance-level annotations for the detection task. The phytoplankton samples are from Jiaozhou Bay in Qingdao, Shandong Province. The phytoplankton in the images is rationally divided into 24 categories by experts and the phytoplankton in all images is categorized to construct an RGB high-resolution phytoplankton detection dataset V1.0 (PMID2019). Compared with the existing datasets, as shown in Figure 1, PMID2019 has the following advantages:

- (i) The resolution of the image is high, i.e. 2040×1536 , which enables the deep network to learn specific details more effectively.
- (ii) Compared with grey-scale images, RGB images are more capable of retaining the effective information of phytoplankton.
- (iii) Each image has instance-level annotations, which can be used for phytoplankton detection tasks.

PMID2019 not only contains images of dead phytoplankton cells, but also has a few images of living cells. Living phytoplankton cells will be observed during the phytoplankton *in situ* observation and it is difficult to capture enough living cell samples in the laboratory environment, because the living cells move rapidly on the microscopic slide. In order to make the detection model have better generalization performance, it is better to establish the mapping between dead cell images and living cell images. To solve this problem, we apply Cycle-GAN (Zhu *et al.*, 2017) to this task, so that both living and dead phytoplankton cell images can be converted to each other without losing their original features. Finally, we get synthetic images of living phytoplankton cells.

The contributions of this article are as follows:

- (i) A phytoplankton microscopic image dataset, PMID2019, is constructed to train the advanced artificial intelligence model for phytoplankton detection.
- (ii) New synthetic images are generated for the *in situ* phytoplankton detection by using Cycle-GAN to migrate the images of phytoplankton dead cells to living cells.
- (iii) We evaluate many state-of-the-art deep learning detection methods on PMID2019, in order to provide new ideas to researchers on phytoplankton microscopic image detection and recognition. The dataset is publicly available on the project page <https://github.com/ouc-ocean-group/PMID2019>.

The rest of this article is organized as follows. Related work section summarizes the related works. Our procedure for constructing the dataset section is the construction procedure of the dataset. Synthetic dataset of living cells section formally introduces our model in detail. Evaluation on PMID2019 section presents the experimental results. Finally, we conclude our work in Conclusion section.

Related work

Non-*in situ* phytoplankton observation method

Optical microscopy is the most traditional method for the detection and analysis of phytoplankton samples. Such detection technique is a method for identifying phytoplankton species based on the morphological characteristics of phytoplankton. It is supplemented by cell counting plates for density counting and has always played an important role in the identification and quantification of phytoplankton (Hallegraeff *et al.*, 1995). However, microscopy testing techniques require testers to have a rich knowledge of phytoplankton taxonomy, which is demanding and time consuming. Scanning electron microscope (SEM) and transmission electron microscope (TEM) are also important tools for phytoplankton research. They can display the fine features of phytoplankton cell surface morphology and internal structure, and are also one of the basic means of phytoplankton identification (Berdach, 2010). Nevertheless, electron microscopy samples which require complex pretreatment processes are time consuming. Flow cytometry FCM has a wide range of applications in the detection of marine phytoplankton (Jonker *et al.*, 1995). Unfortunately, flow cytometry cannot effectively identify target cells with weak fluorescent labelling, and there is a possibility of missed recording, and the high cost of the instrument limits the wide application of flow cytometry in the detection of phytoplankton. The above methods are all non-*in situ* phytoplankton detection methods.

In situ phytoplankton observation methods

Acoustic systems, such as Acoustic Doppler Current Profiler, Multifrequency Hydroacoustic Probing System, and wideband sonar have many advantages for plankton detection. Sound waves are less affected by underwater environments. The characteristics of underwater propagation are better than for visible light and electromagnetic waves. Therefore, sound waves can be used for long-distance positioning. However, its poor ability to distinguish underwater organisms, unapparent detail features, and inaccurate positioning affects the reliability and accuracy of observation (Warren *et al.*, 2001). Chlorophyll fluorescence instrument for phytoplankton detection is the most mature, diverse, and widely used *in situ* observation device for marine organisms. Its main disadvantage is that it can only detect auto-fluorescent organisms (Kolber and Falkowski, 1993). Because the advent of optical microscope, it has become an important tool for basic micro-organism research. In 2004, Yu (Yu *et al.*, 2004) developed an “underwater automated digital microscope imager” based on optical microscopes that can be directly placed in water to achieve automated shooting of *in situ* plankton images. However, the instrument is only suitable for high concentration conditions. As early as 1992, the first *in situ* automated identification device, Video Plankton Recorder, was produced and has become a pioneer in modern *in situ* plankton imaging devices (Sullivan-Silva and Forbes, 1992). For small and microplankton, Olson *et al.* used the Imaging FlowCytobot for long-term monitoring of microzooplankton and phytoplankton from 10 to 100 μm (Olson and Sosik, 2007). Similar systems are also available, including FlowCAM, CytoSense, CytoBuoy, and CytoSub, which enable the acquisition of information from micro to small phytoplankton, providing an effective method for *in situ* observation of full-grained phytoplankton.

Existing phytoplankton datasets

There are two categories of existing phytoplankton datasets. One is related to abundance, biomass, and composition, and the other is the phytoplankton microscopic image dataset.

COPEPOD's global plankton database (O'Brien, 2005) provides plankton and ecosystem researchers with an integrated dataset of quality-reviewed, globally distributed plankton abundance, biomass, and composition data. In addition to data distribution maps, COPEPOD offers a variety of text and graphical content summaries and searching options. The Belgian Phytoplankton Database is a comprehensive data collection comprising quantitative phytoplankton cell counts from multiple research projects conducted since 1968. The collection is focused on the Belgian part of the North Sea, but also includes data from the French and the Dutch part of the North Sea. The database includes almost 300 unique sampling locations and more than 3000 sampling events resulting in more than 86 000 phytoplankton cell count records (Nohe et al., 2018).

WHOI-Plankton (Orenstein et al., 2015) is a large-scale, fine-grained visual recognition dataset for plankton classification, which comprises over 3.4 million expert-labelled images across 70 classes. The labelled image set is compiled from over 8 years of near continuous data collection with the IFCB at the Martha's Vineyard Coastal Observatory since 2006. But the images in this dataset only have image-level labels, which can only be used to classify different plankton. Kaggle-Plankton (Li and Cui, 2016), which consists of 30 336 plankton images of 121 classes is also a dataset for plankton classification. The images in this dataset are low-resolution grey images, which are not conducive to getting the detail features. ZooScan (Gorsky et al., 2010) is a zooplankton dataset including 20 classes. The images are also low-resolution grey images.

Researchers from Xiamen University use laboratory specimens to collect common phytoplankton samples from coastal areas of China. They collect images of morphological characteristics of algae cells on different sides using digital microphotography techniques, and combine them with textual data to establish a database of common phytoplankton network searches in China's coastal areas. At present, the database includes 144 species of common phytoplankton in China, with 704 characteristic images, mainly diatoms (93 species) and dinoflagellate (40 species). This database consists of the main characteristic parameters and ecological distribution characteristic parameters of cells. These parameters are used for detailed information retrieval, rather than directly identifying images. They also constructed a digital microscopic image database of common marine phytoplankton species in China. The database contains 3239 images from 241 species of phytoplankton, including 168 species of diatoms, 70 species of dinoflagellates, and 3 other phytoplankton. However, the resolution of each image is very low.

Domain adaptation

Recently, transfer learning (Sun et al., 2018) and domain adaptation methods have been proposed to mitigate the domain gap. These methods can be divided into three categories. The first one is to introduce different learning schema to align the source and target domains. Inspired by the kernel two-sample test (Gretton et al., 2008), Maximum Mean Discrepancy is applied to reduce distribution shift in various methods (Ghifary et al., 2014; Long et al., 2017). The second category is the adversarial-based approach. A domain discriminator is leveraged to encourage the domain confusion by an adversarial objective. Generative adversarial networks are widely utilized to learn domain-invariant features as well to generate target source (Dong et al., 2019).

The third category is reconstruction-based. The reconstruction is obtained by an encoder–decoder or a GAN discriminator such as Dual-GAN (Yi et al., 2017), Cycle-GAN (Zhu et al., 2017), and Disco-GAN (Kim et al., 2017). Self-ensembling is also utilized for visual adaptation problems (French et al., 2017). Wang et al. propose a combined model to learn feature and class jointly invariant representation (Wang et al., 2018). In Peng et al. (2018), the researchers propose a new deep learning approach, Moment Matching for Multi-Source Domain Adaptation, which aims to transfer knowledge learned from multiple labelled source domains to a un-labelled target domain.

Our procedure for constructing the dataset

In order to construct the detection dataset of phytoplankton, we first capture tens of thousands of phytoplankton microscopic images using an optical microscope in the laboratory environment. Then we divide all the phytoplankton cells into 24 categories according to the advice of marine biologists. Finally, we label all the images to localize and classify each object as ground-truth.

Image acquisition

In this article, an Olympus BX53 was used to collect the microscopic images of phytoplankton. The samples were taken from the sea area of Qingdao Jiaozhou Bay, and formaldehyde solution was added to fix the phytoplankton morphology. The slide was prepared by pipetting an appropriate amount of phytoplankton formaldehyde solution then placed on the stage. The magnification of the objective lens was set to 20 times, the eyepiece was 10 times, and the overall magnification was 200 times. The stage was moved in an S-shape from the upper right corner of the slide to obtain a microscopic image in all fields of view. And the image in the field of view with phytoplankton was collected. In order to increase the richness of our dataset, we collected images of different illumination conditions in the same field of view. On the other hand, we acquired images obtained by fine-focusing spiral adjustment. The image resolution is 2040×1536 . Raw images have three uncompressed colour channels. Figure 2 shows the images obtained by adjusting different focal lengths through a fine focus screw, and the images in the second row are the magnification of the image in the blue box. Figure 3 shows images of different illumination conditions in the same field of view.

A few living cell images in the dataset

Most of the samples in our dataset are dead cells. The reason is that the cells prepared with formaldehyde can maintain a fixed shape for image acquisition. Meanwhile, we still collect a few living cell samples for dynamic change process. However, it is a difficult task to obtain living phytoplankton because they tend to move very fast. So it is very hard to obtain the images. There are only 217 living cell images including 10 different categories in our dataset. And each category only consists of a few images. Figure 4 shows some images of living cell samples. Compared with the dead cell images in Figures 2 and 3, the living cells are all yellowish because of the plastid inside. In Figure 5, three different categories, *Pleurosigma pelagicum*, *Ceratium furca*, and *Ceratium trichoceros*, are shown from top to bottom. Each row of the images is from the same phytoplankton with different motion states. The task of collecting images of living cells is particularly complex and difficult. Therefore, in order to achieve the *in situ* observation of phytoplankton, it is necessary to construct a

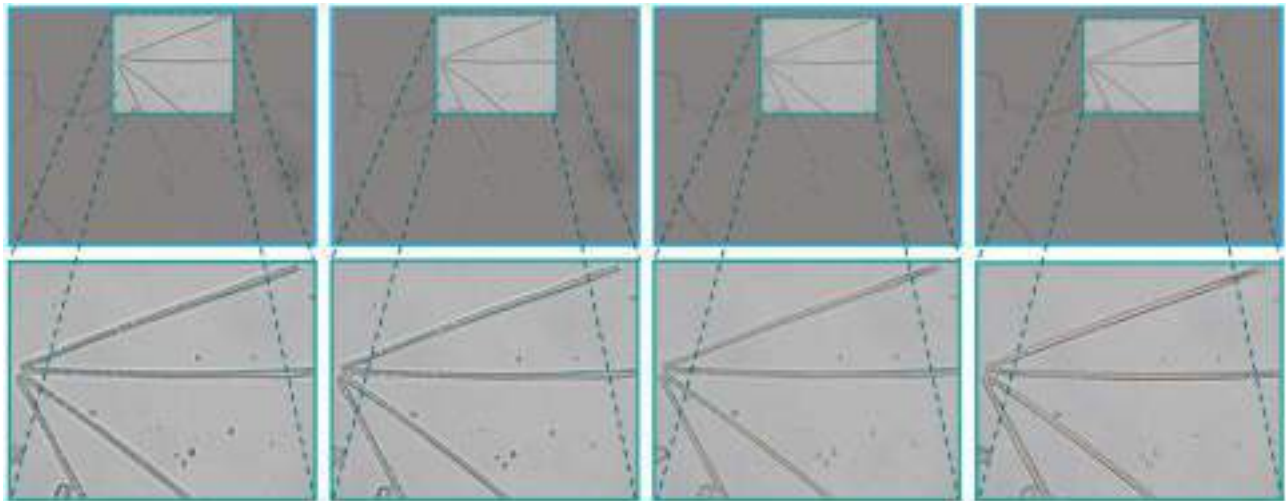


Figure 2. Images of the same view obtained by adjusting different focal lengths through a fine focus screw.

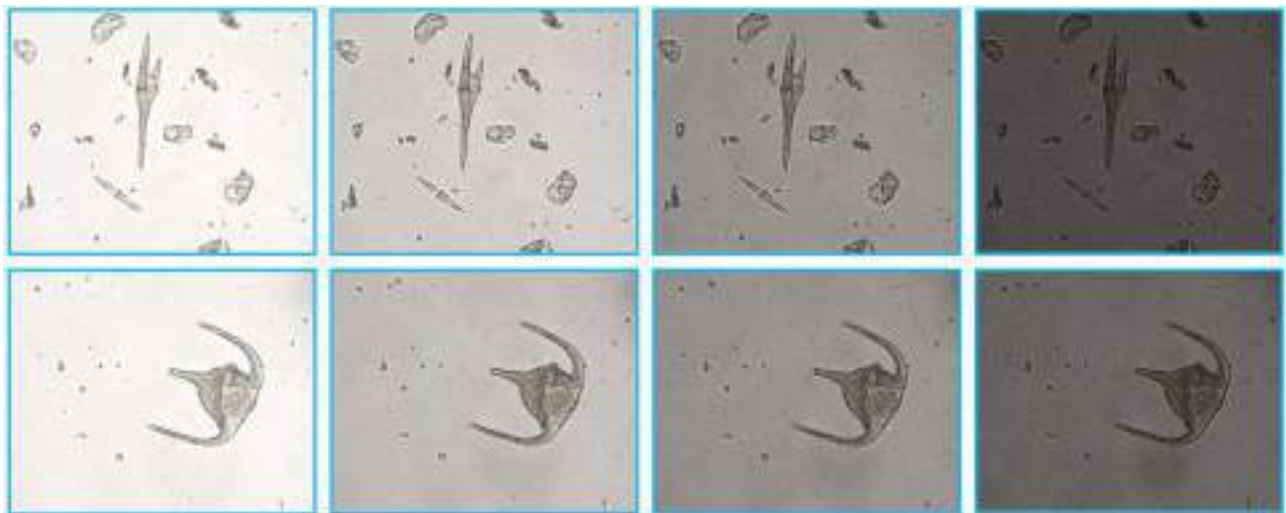


Figure 3. Eight illustrative images of the database: images of different lighting conditions.

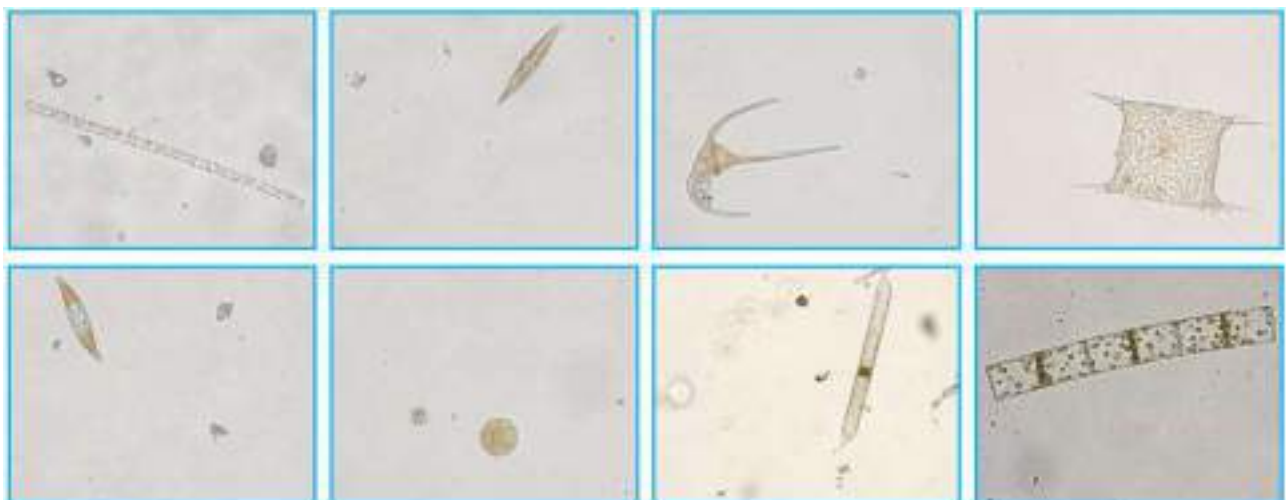


Figure 4. The living cell images in our dataset belonging to eight categories, from left to right, top to bottom: *Skeletonema*, *Navicula*, *Ceratium furca*, *Biddulphia*, *Pleurosigma pelagicum*, *Coscinodiscus*, *Rhizosolenia*, and *Guinardia flaccida*.



Figure 5. Each row belongs to the same phytoplankton with different motion forms, from top to bottom, *Pleurosigma pelagicum*, *Ceratium furca*, and *Ceratium trichoceros*.

synthetic dataset of phytoplankton living cells. We will introduce the details in Synthetic dataset of living cells section.

Object category

The sample collection work took a total period of 2 months and we finally collected 10 819 images, each of which contains a number of phytoplankton cells (i.e. averaging 3 cells per image). Under the guidance of marine ecology experts, we divided all the collected phytoplankton cells collected into the 24 categories including *Chaetoceros*, *Eucampia*, *Skeletonema*, *Coscinodiscus*, *Thalassionema nitzschioides*, *Guinardia flaccida*, *Ceratium furca*, *Ceratium fusus*, *Bacteriastrum*, *Ceratium trichoceros*, *Thalassionema frauenfeldii*, *Dinophysis caudata*, *Biddulphia*, *Helicotheca*, *Pleurosigma pelagicum*, *Ceratium tripos*, *Navicular*, *Ditylum*, *Protoperidinium*, *Rhizosolenia*, *Detonula pumila*, *Coscinodiscus flank*, *Corethron*, and *Ceratium carriense*. The amount of each category is shown as a histogram in Figure 6 where *Chaetoceros* contains the most samples and *Ceratium carriense* the least. The extreme imbalance of the samples increases the difficulty of detection. However, according to the experiment results in Evaluation on PMID2019 section, the categories which only have a few samples also have a high detection accuracy. In the meantime, it can be used to evaluate the performance of the methods for solving sample imbalance.

Annotations

Each phytoplankton target in the captured microscopic images has been manually annotated to form the ground-truth. The ground-truth labels are important for the supervised learning techniques in machine learning.

We used the ground-truth generation tool named LabelImg (Tzutalin, 2015) to label the images. Six human annotators were asked to overlay a bounding box tightly around the object one by one in the image. Each object was given a category it belongs to. Bounding boxes are used to locate the objects in an image. Then LabelImg will automatically generate an Extensible Markup Language file of the image to save the coordinates (in pixels) of its four corners and its class label. Each annotator spent around 30 s to draw a bounding box in an image. Figure 7 shows some images with bounding boxes. It can be seen that each object in the image is tightly surrounded by a blue bounding box.

Comparison with the existing datasets

To the best of our knowledge, PMID2019 is the first high resolution dataset for the detection of phytoplankton. As shown in Figure 1, we show some examples of the existing dataset according to their original scale of image size. The images from the various datasets are scaled to show the relative resolutions of the datasets. ZooScan is a zooplankton dataset including 20 classes with low-resolution grey images. It can only be used to classify zooplankton. WHOI-Plankton is a dataset for plankton classification consisting of 3.4 million labelled grey images divided into 70 classes. Kaggle-Plankton contains low-resolution grey images for plankton classification. The same as the others, ASLO can only be utilized for plankton classification. Compared with these existing datasets, PMID2019 is the first dataset with high-resolution colour images for phytoplankton detection. The resolution of each image is 2040×1536 , much higher than the images in the compared datasets. Our dataset also has different lighting conditions to simulate the real *in situ* environment and help train a robust detection model.

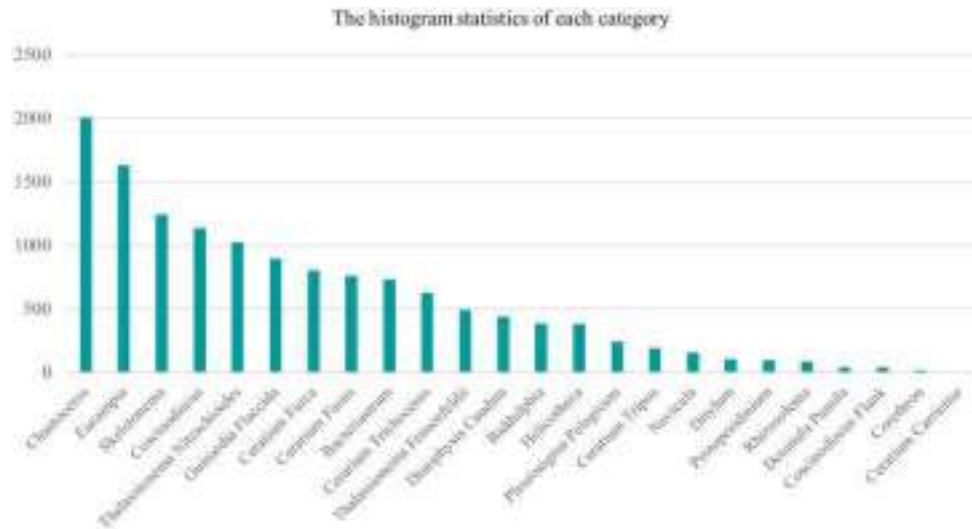


Figure 6. A histogram statistics of the dataset. The phytoplankton cells are divided into 24 categories as shown above. From left to right, the value progressively becomes smaller and the amount of each category is extremely unbalanced.

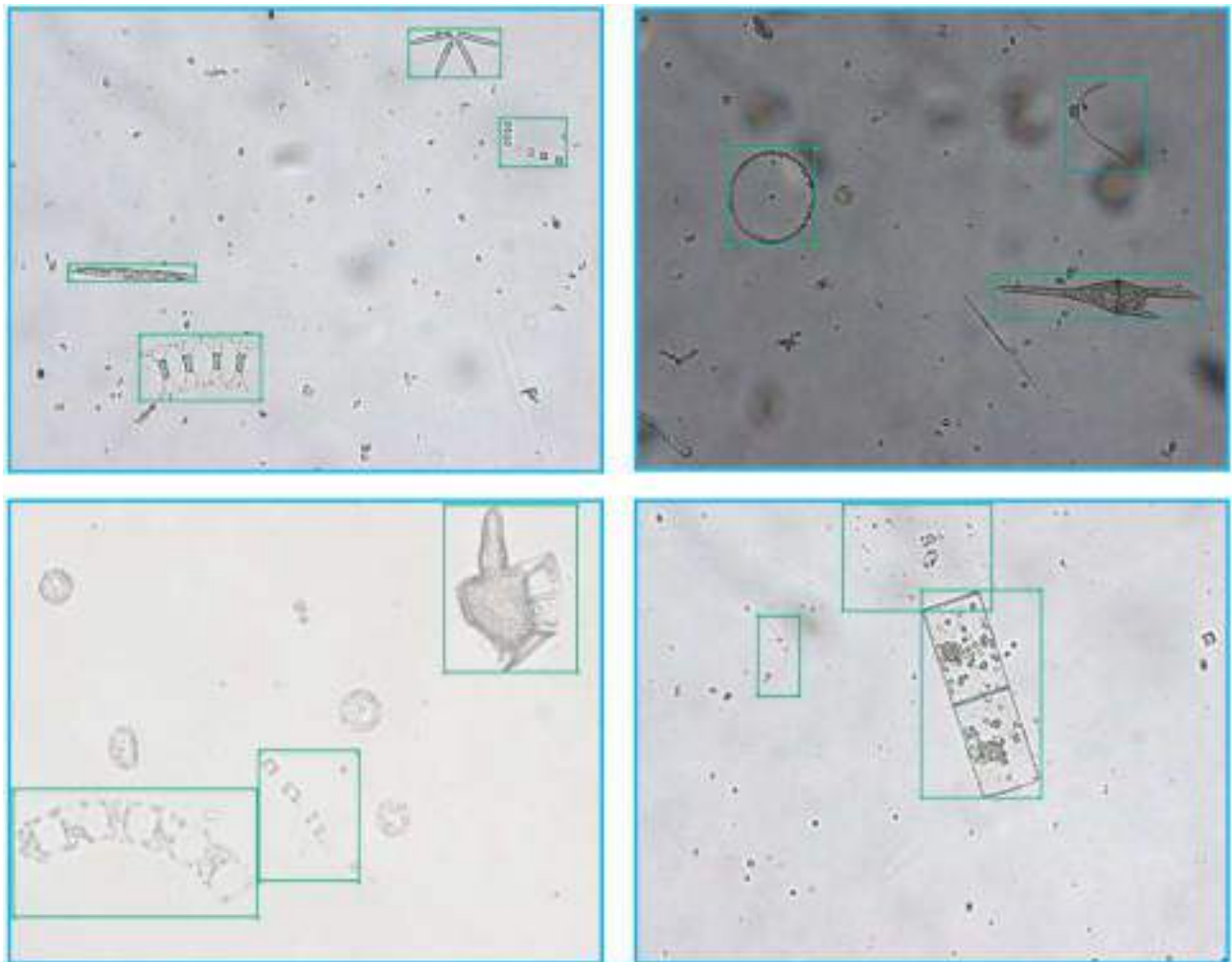


Figure 7. Some images with bounding boxes.

We also cut out all the bounding boxes in the dataset to generate a classification dataset, which can also be able to achieve excellent performance in the classification task.

Synthetic dataset of living cells

In our dataset, just 2% of the images are living cells, the rest are dead phytoplankton treated with formaldehyde. However, living phytoplankton cells will be acquired during the *in situ* observation procedures. To generalize the dataset to *in situ* applications, we use Cycle-GAN (Zhu et al., 2017) to achieve the domain migration between dead and living cell, so as to generate the corresponding living cell images from the original dead cell images in order to build a synthetic dataset.

Before the introduction of Cycle-GAN, we will give a brief explanation of GAN (Generative Adversarial Nets). It is a framework for estimating generative models via an adversarial process, in which two models are simultaneously trained: a generative model G that captures the data distribution, and a discriminative model D that estimates the probability that a sample comes from the training data rather than G . Cycle-GAN is an approach for learning to translate an image from a source domain X to target domain Y . The space of dead cell images constitutes the source domain X (the grey box in Figure 8) and the limited living cell belongs to the target domain Y (the yellow box in Figure 8). Our goal is to learn the mapping functions between the two domains X and Y , given training samples of real dead cell images $\{x_i\}_{i=1}^N$, where $x_i \in X$, and real living cell images $\{y_i\}_{i=1}^M$, where $y_i \in Y$. The data distribution is denoted as $x \sim p_{\text{data}}(x)$ and $y \sim p_{\text{data}}(y)$. As shown in Figure 8, there are two mappings, i.e. $GL: X \rightarrow Y$ and $GD: Y \rightarrow X$, in addition, two adversarial discriminators DD and DL , where DD aims to distinguish between real dead cell images $\{x\}$ and translate fake dead cell images $\{GD(y)\}$. In the same way, DL aims to discriminate between real living cell images $\{y\}$ and translates fake living cell images $\{GL(x)\}$. The grey lines mean the process of translating dead cell images to living cell images, and the yellow lines mean the opposite direction from living cell images to dead cell images. The solid ones mean transferring real cell images

into fake cell images, and the dotted ones mean the process from fake cell images into real cell images.

The objective function, as shown in Equation (1), contains two adversarial losses (Goodfellow et al., 2014) of both mapping functions in Equations (2) and (3) and one cycle consistency loss (Zhu et al., 2017) \mathcal{L}_{cyc} in Equation (4) to guarantee that the learned function can map an individual input x_i to a desired output y_i . Therefore, the objective functions are as follows:

$$\mathcal{L}(GD, GL, DD, DL) = \mathcal{L}_{\text{GAN}}(GD, DL, X, Y) + \mathcal{L}_{\text{GAN}}(GL, DD, Y, X) + \lambda \mathcal{L}_{\text{cyc}}(GD, GL) \tag{1}$$

$$\mathcal{L}_{\text{GAN}}(GD, DL, X, Y) = \mathbb{E}_{y \sim p_{\text{data}}(y)} [\log DL(y)] + \mathbb{E}_{x \sim p_{\text{data}}(x)} [\log(1 - DL(GD(x)))] \tag{2}$$

$$\mathcal{L}_{\text{GAN}}(GL, DD, Y, X) = \mathbb{E}_{x \sim p_{\text{data}}(x)} [\log DD(x)] + \mathbb{E}_{y \sim p_{\text{data}}(y)} [\log(1 - DD(GL(y)))] \tag{3}$$

$$\mathcal{L}_{\text{cyc}}(GD, GL) = \mathbb{E}_{x \sim p_{\text{data}}(x)} [\|GL(GD(x)) - x\|_1] + \mathbb{E}_{y \sim p_{\text{data}}(y)} [\|GD(GL(y)) - y\|_1] \tag{4}$$

The generative network contains two stride-2 convolutions, six residual blocks, and two half stride convolutions. Similar to Johnson et al. (2016), we use instance normalization. For the discriminator networks we use 70×70 PatchGANs (Isola et al., 2016; Ledig et al., 2017), which aims to determine whether 70×70 overlapping image patches are real or fake. The parameters of such a patch-level discriminator architecture are fewer than a full-image discriminator. We use the settings as suggested by Zhu et al. (2017). For all experiments, we set $\lambda = 10$ in Equation (1). Adam solver is utilized and the batch size is set to 1. The size of the input image is 256×256 . The network is trained from scratch for 200 epochs and the learning rate is set to 0.0002. We keep the same learning rate for the first 100 epochs then linearly decay the rate to zero for the remaining 100 epochs.

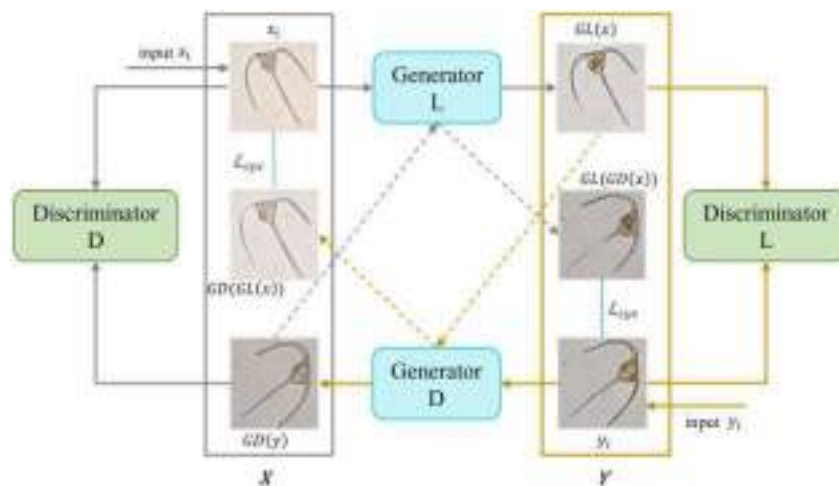


Figure 8. The migration model contains two mapping functions $GL: X \rightarrow Y$ and $GD: Y \rightarrow X$, and the associated adversarial discriminators DD and DL . The grey box is the source domain X , the yellow box is the target domain Y . There is a cycle consistency loss \mathcal{L}_{cyc} to guarantee both GD can transform the fake living cell images generated by GL into dead cell images as illustrated in Figure 8 by grey dotted lines, and GL can transform the fake dead cell images generated by GD into living cell images as shown by yellow dotted lines.

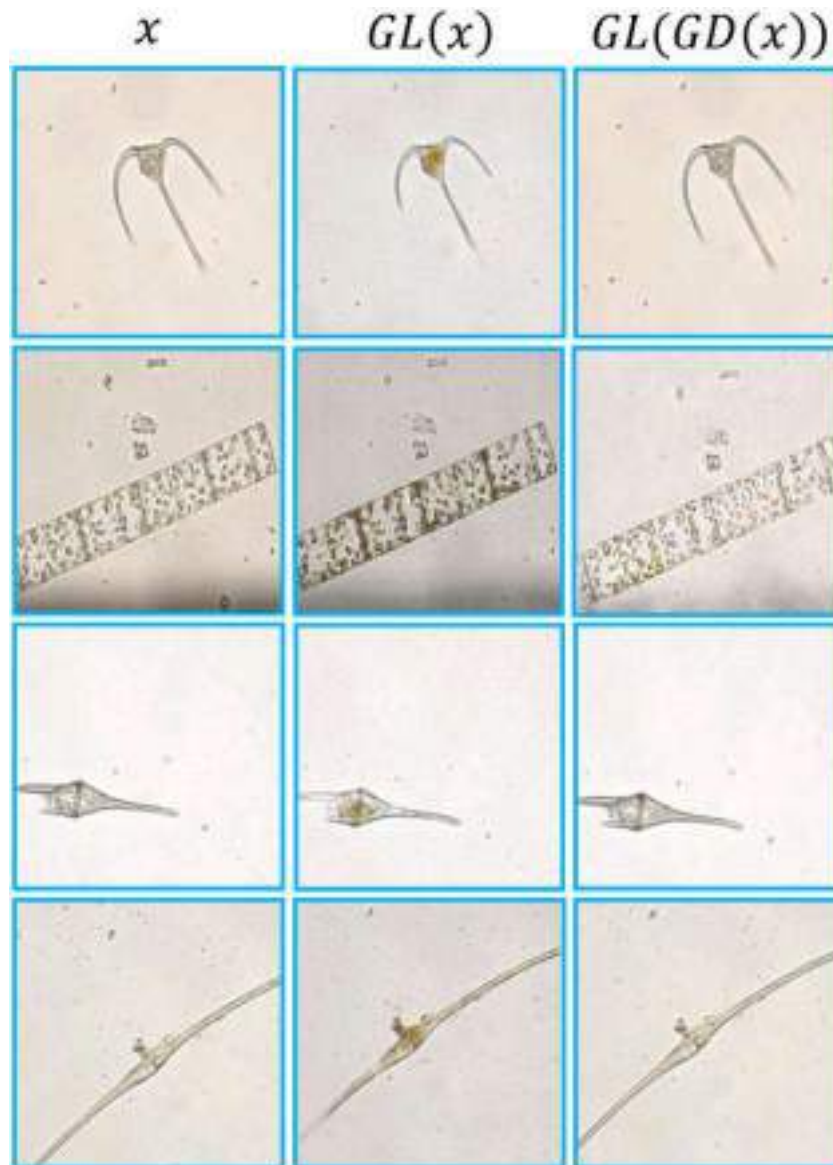


Figure 9. The input images x are real dead cell images from the source domain X . The images in the middle column are output images $GL(x)$, which have been translated into the target domain Y . The images in the right column are reconstructed images $GD(GL(x))$, which match closely to the input images x . From top to the bottom: *Ceratium trichoceros*, *Guinardia flaccida*, *Ceratium furca*, and *Ceratium fusus*.

Figure 9 shows the results of various phytoplankton cells, including *Ceratium trichoceros*, *Guinardia flaccida*, *Ceratium furca*, and *Ceratium fusus*.

With Cycle-GAN, we can finally migrate all the dead cell images into corresponding living cell images. It helps us to construct a more comprehensive, adequate, and reasonable synthetic dataset that can be utilized in the *in situ* observation scenarios.

A set of experiments are carried out to prove the effectiveness of the utilization of Cycle-GAN. First, we split the captured images into real images of dead and living cells. Then we use Cycle-GAN to generate 1000 images of 5 categories to make a Synthetic living cells dataset. Third, we train a typical detector, Faster R-CNN, on Real dead cells, Real living cells, and PMID2019 consisting of Synthetic images separately. Finally, we test each model for detection and classification on the Real living cells of PMID2019 and we get average precision values of 75%.

The detection results of the five categories are shown in Table 1. Although the amount of Real dead cells is larger than Real living cells, the performance of the latter is better than the former, which illustrates that living cell samples play an important role in the *in situ* detection process. The better performance of the model trained on PMID2019 consisting of Synthetic data depends on a larger dataset with more living cell samples.

Evaluation on PMID2019

The main purposes of evaluating baseline methods on the phytoplankton microscopic image dataset are as follows. First, we want to investigate the difficulties of the state-of-the-art detection methods on the microscopic dataset to make a benchmark. Second, we want to provide reference evaluation results to researchers on phytoplankton microscopic image detection and computer vision. Third, we want to find out the shortcomings of

Table 1. The real living cell detection results of Faster R-CNN models trained with real dead cells, real living cells, and PMID2019 consisting of synthetic data separately.

Phytoplankton	Real dead cells from PMID2019	Real living cells from PMID2019	PMID2019
<i>Navicula</i>	0.3636	0.8831	0.9900
<i>Ceratium furca</i>	0.8695	0.9051	0.9992
<i>Ceratium trichoceros</i>	0.8972	0.9091	0.9870
<i>Guinardia flaccida</i>	0.8329	0.8894	0.9091
<i>Ceratium fusus</i>	0.7091	0.9091	0.9910

existing state-of-the-art detection methods for microscopic images. In the following, we first introduce several state-of-the-art object detection approaches. All code can be downloaded from our project page: <https://github.com/ouc-ocean-group/PMID2019>.

Faster R-CNN

Faster R-CNN (Ren et al., 2015) is a state-of-the-art object detection network that is different from the earlier detectors, which depend on region proposal algorithms like SPPnet (He et al., 2015) and Fast R-CNN (Girshick, 2015). A region proposal network (RPN), which shares full-image convolutional features with detection networks is introduced in Faster R-CNN. The RPN is a fully convolutional network (Shelhamer et al., 2014), which can be trained end-to-end specifically for generating detection proposals instead of the previous algorithms like Selective Search (Uijlings et al., 2013) and EdgeBoxes (Zitnick and Dollár, 2014). RPNs are trained to efficiently predict region proposals with a wide range of scales and aspect ratios. Therefore, “anchor” boxes are introduced as references at different scales and aspect ratios.

The structure of Faster R-CNN can be divided into four stages. The first stage of Faster R-CNN is a backbone consisting of convolutional layers that are used for feature extraction. They are taken from a pre-trained image classification network, one trained on the ImageNet dataset. Common choices are VGG-16, ResNet-50, or ResNet-101. In this article, both ResNet-101 and VGG-16 are used. The second stage is RPN. After receiving the feature maps from the first stage, the RPNs will generate region proposals then transfer to the next stage. We use a 3×3 convolutional layer followed by two sliding 1×1 convolutional layers for regression and classification, respectively. The third stage is ROI (Region of Interest) Pooling. ROI pooling treats the features extracted by the shared backbone as an image and crops rectangular regions corresponding to the regions predicted by the RPN. The final stage consists of a classifier that identifies the type of object and a regressor that predicts final bounding box refinements to improve the accuracy of the bounding box. In the meantime, non-maximum suppression (Hosang et al., 2017) is used to filter the predictions from both the RPN and the final bounding box predictions. We randomly initialize all new layers by drawing weights from a zero-mean Gaussian distribution with standard deviation 0.01. All other layers are initialized by pre-training a model for ImageNet classification. We use a learning rate of 0.001 for 60k mini-batches, and 0.0001 for the rest on our dataset. The other implementation details are as described in Ren et al. (2015).

Feature pyramid network

Feature pyramids built upon image pyramids form the basis of a standard solution of recognizing objects at vastly different scales

(Adelson et al., 1983). FPN (feature pyramid network) is a clean and simple framework for building feature pyramids inside convolutional networks (Lin et al., 2017a). Feature pyramids with marginal extra cost are constructed by exploiting the inherent multi-scale, pyramidal hierarchy of deep convolutional networks. FPN develops a top-down architecture with lateral connections to build high-level semantic feature maps at all scales. It can be trained end-to-end with all scales and is used consistently at train/test time.

FPN takes a single-scale image with a random size as input, and outputs proportionally sized feature maps at multiple levels, in a fully convolutional way. The process is independent of the backbone convolutional architectures (e.g. Krizhevsky et al., 2012; He et al., 2016; Yan et al., 2015). FPN combines the low-resolution high-level features from the later layers of the convolutional backbone with higher resolution lower-level features drawn from the backbone via lateral connection. This helps the network resolve high resolution structures while retaining the semantic richness of the high-level features from later layers in the backbone. Using FPN in a basic Faster R-CNN system achieves state-of-the-art results. The input image is resized such that its shorter side has 800 pixels. The learning rate is 0.02 for the first 30k mini-batches and 0.002 for the rest. The other settings are as described in Lin et al. (2017a). Both FPN with Faster R-CNN and Faster R-CNN are two-stage region-based detectors.

Single shot multibox detector

Although the above detectors achieve high accuracies, they are too computationally intensive for embedded systems and too slow for real-time applications. SSD (single shot multibox detector; Liu et al., 2016) is a method for detecting objects in images using a single deep neural network. It is the first deep network-based object detector that does not resample pixels or features for bounding box hypotheses and achieves good results.

This method utilizes a small convolutional filter to predict object categories and offsets in bounding box locations. In order to perform detection at multiple scales, it uses separate predictors for different aspect ratio detections, and applies these predictors to multiple feature maps from the later stages of a network. SSD can be trained in an end-to-end way and achieves high accuracy. We fine-tune the resulting model using SGD with initial learning rate 10^{-3} , 0.9 momentum, 0.0005 weight decay, and batch size 32. Then, after 40k iterations, continue training for 10k iterations with 10^{-4} and 10^{-5} . The other implementation details are as described in Liu et al. (2016). It is faster and more accurate than the present state-of-the-art detectors such as (YOLO) for single shot detectors, and even as accurate as slower detectors such as Faster R-CNN.

YOLOv3

YOLO (You Only Look Once) utilizes an end-to-end single neural network to predict bounding boxes and class probabilities directly from full images in one evaluation. It frames object detection as a regression problem to spatially separate bounding boxes and associated class probabilities. However, YOLO is not effective enough for small objects in the image, and the generalization ability would be very weak when the same object has a new uncommon aspect ratio. Especially, because of the limitation of the loss function, location error is the main problem affecting detection results. Therefore, the improved versions, YOLOv2,

YOLO9000 (Redmon and Farhadi, 2017) and YOLOv3 (Redmon and Farhadi, 2018) were gradually proposed.

YOLOv3 uses a few tricks to improve training and increase performance, including multi-scale predictions, a better backbone classifier, and an effective loss function. The backbone network is Darknet-53 with much deeper convolutional networks and has some shortcut connections to avoid gradient disappearance. In the prediction period, YOLOv3 extracts features from multiple scales using a similar concept to feature pyramid networks. The deep features provide semantic information, and the shallow features can provide fine-grained information. During training, binary cross-entropy loss is used for the classification.

RetinaNet

The detectors with the highest accuracy are based on a two-stage, proposal-driven mechanism. The first stage generates a sparse set of region proposals and the second stage classifies each region proposal using a convolutional neural network. In contrast, one-stage detectors are applied over a regular, dense sampling of object locations, scales, and aspect ratios. To help the one-stage detectors achieve similar accuracy as the two-stage approaches, a novel Focal Loss function is proposed to address class imbalance. Therefore, RetinaNet (Lin et al., 2017b) was designed and trained to evaluate the effectiveness of this loss.

The loss function is a dynamically scaled cross-entropy loss. When confidence in the correct class increases, the scaling factor decays to zero. The scaling factor can automatically down-weight the contribution of easy examples during training and rapidly focus the model on hard examples. RetinaNet is a single, unified network composed of a backbone network, which is utilized to compute convolutional feature maps and two task-specific sub-networks. In this work, we experiment with the ResNet-50

backbone, which is pre-trained on ImageNet. The model is trained for 90k iterations with an initial learning rate of 0.01, which is then divided by 10 at 60k and again at 80k iterations. Weight decay of 0.0001 and momentum of 0.9 are used. The other settings are the same as described in Lin et al. (2017b). One subnetwork is to perform convolutional object classification; the other is to perform convolutional bounding box regression. This approach is simple and highly effective and achieves state-of-the-art accuracy and speed.

Phytoplankton detection

In our experiments, we evaluate state-of-the-art baseline detectors: Faster R-CNN (Ren et al., 2015), FPN, SSD (Liu et al., 2016), YOLOv3 (Redmon et al., 2016; Redmon and Farhadi, 2018), and RetinaNet (Lin et al., 2017b) on our dataset. We separate the dataset into two parts, 50% images of training set, 50% images of test set. Unless otherwise noted, we leveraged their default settings. The first group of the detection models in Table 2 is two-stage region-based detectors, the second group one-stage region-free detectors. Six classes are selected based on the amount to show their detection results as displayed in Table 2. We utilize average precisions (APs) at different intersection over union (IoU) where a predicted bounding box is correct if its IoU with the ground-truth bounding box is higher than 0.5 or 0.75 and Aps for different object sizes as the main evaluation metrics.

Figure 10 shows the qualitative prediction results of the previous methods. The original images are in the first row, the second to the fourth rows are sequentially the prediction results of Fast R-CNN, SSD, and YOLOv3. The first 4 columns in Figure 10 are simple images of our dataset. Different scenarios are shown in Figure 10, various lighting conditions from bright to dark like columns e, f, g, and h, complex background like columns g, h, i,

Table 2. The detection results on PMID2019 with state-of-the-art methods.

Detection Methods	Backbone	AP ₅₀ (%)	AP ₇₅ (%)	CHAE	SKEL	DICA	NAVI	RHIZ	COFL
Faster R-CNN	ResNet-101	91.26	87.65	89.55	88.61	96.30	86.58	83.10	70.27
Faster R-CNN	VGG16	90.54	86.95	88.99	85.98	99.13	84.78	82.83	62.79
FPN	RssNet-101	92.68	89.19	80.89	80.12	99.95	73.89	75.39	70.95
SSD	ResNet-101	88.12	84.66	77.81	76.06	90.91	67.31	75.95	70.75
YOLOv3	DarkNet-19	93.10	82.81	79.11	67.29	96.94	61.40	65.10	50.11
RetinaNet	ResNet-50	89.25	88.82	88.12	87.18	98.77	72.70	67.81	79.28

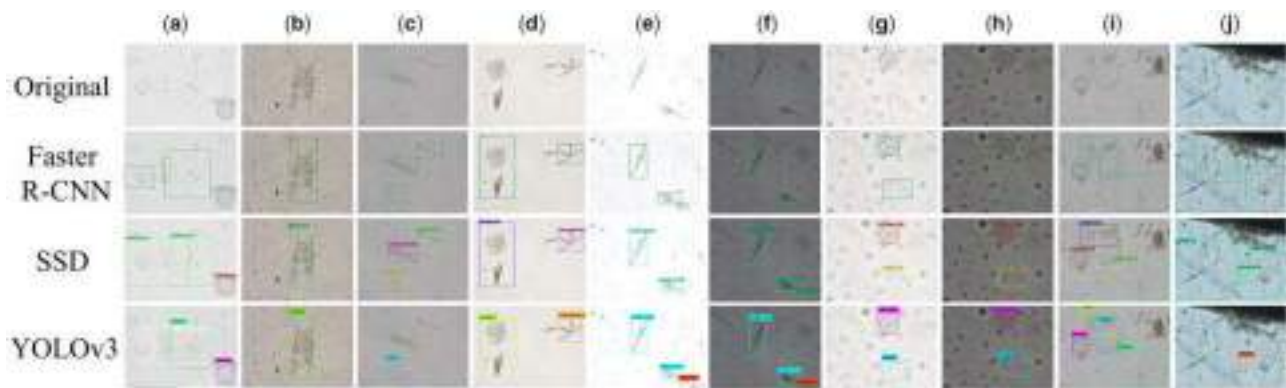


Figure 10. The original images are in the first row, the second to the fourth rows are sequentially the prediction results of Fast R-CNN, SSD, and YOLOv3.

and j, and overlap like columns i and j. Compared with the other two detectors, the prediction results of Fast R-CNN are the best, which can accurately predict the location and class of all the phytoplankton cells in the images.

Conclusion

In this article, we introduced a new phytoplankton microscopic image dataset, which contains 10 819 phytoplankton microscopic images of 24 different classes. Each object in the image has been manually labelled with a bounding box and category of ground-truth. In order to generalize the dataset for *in situ* applications, we further utilize Cycle-GAN to achieve the domain migration between dead and living cell samples. We build a synthetic dataset to generate the corresponding living cell samples from the original dead cell ones. The PMID2019 dataset can not only be used to assess and evaluate the performance of the state-of-the-art phytoplankton detection algorithms on microscopic images, but also can particularly benefit the development of phytoplankton microscopic vision technology in the future. Marine biologists can use PMID2019 to train a detection model to help them count, detect, and classify phytoplankton automatically. Therefore, it is able to gain great benefits for scientific research. Moreover, the resolution of the images in our dataset is large enough for marine biologists to study the detailed features of both phytoplankton cells. The reported performances of some state-of-the-art object detection algorithms may provide new ideas for the monitoring of marine ecosystems.

Acknowledgements

We would like to express our sincere appreciation to the anonymous reviewers for their insightful comments, which have greatly aided us in improving the quality of the article. This work was supported by the National Natural Science Foundation of China (U1706218, 61971388), the Key Research and Development Program of Shandong Province (GG201703140154); and Applied Basic Research Programs of Qingdao (18-2-2-38-jch). This work got the GPU computation support from Center for High Performance Computing and System Simulation, Pilot National Laboratory for Marine Science and Technology (Qingdao).

References

- Adelson, E. H., Anderson, C., Bergen, J. J., Burt, P., and Ogden, J. 1983. Pyramid methods in image processing. *RCA Engineer*, 29: 33–41.
- Badjatiya, P., Gupta, S., Gupta, M., and Varma, V. 2017. Deep learning for hate speech detection in Tweets. *In Proceedings of the 26th International Conference on World Wide Web Companion*, pp. 759–760.
- Berdach, J. T. 2010. In situ preservation of the transverse flagellum of *Peridinium cinctum* (Dinophyceae) for scanning electron microscopy. *Journal of Phycology*, 13: 243–251.
- Charlson, R. J., Lovelock, J. E., Andreae, M. O., and Warren, S. G. 1987. Oceanic phytoplankton, atmospheric sulphur, cloud albedo and climate. *Nature*, 326: 655–661.
- Deng, L. 2014. *Automatic Speech Recognition: A Deep Learning Approach*. Springer, London, Heidelberg, New York, Dordrecht.
- Deng, L., and Yu, D. 2014. *Deep learning: methods and applications*. Foundations & Trends in Signal Processing, 7: 197–387.
- Dong, J., Wang, L., Liu, J., and Xin, S. 2019. A procedural texture generation framework based on semantic descriptions. *Knowledge-Based System*, 163: 898–906.
- Fayek, H. M., Lech, M., and Cavedon, L. 2017. Evaluating deep learning architectures for speech emotion recognition. *Neural Networks*, 92: 60–68.
- French, G., Mackiewicz, M., and Fisher, M. 2017. Self-ensembling for visual domain adaptation. *arXiv preprint arXiv:1706.05208*.
- Ghifary, M., Kleijn, W. B., and Zhang, M. 2014. Domain adaptive neural networks for object recognition. *Pacific Rim International Conference on Artificial Intelligence*, pp. 898–904.
- Girshick, R. 2015. Fast R-CNN. *In Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440–1448.
- Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. *et al.* 2014. Generative adversarial nets. *In Advances in Neural Information Processing Systems*, Sarawak, pp. 2672–2680.
- Gorsky, G., Ohman, M. D., Picheral, M., Gasparini, S., Stemmann, L., Romagnan, J. B., Cawood, A. *et al.* 2010. Digital zooplankton image analysis using the ZooScan integrated system. *Journal of Plankton Research*, 32: 285–303.
- Gretton, A., Borgwardt, K., Rasch, M. J., Scholkopf, B., and Smola, A. J. 2008. A Kernel method for the two-sample problem. *In Advances in Neural Information Processing Systems*, pp. 513–520.
- Hallegraeff, G. M., Anderson, D. M., and Cembella, A. 1995. *Manual on harmful marine microalgae*. IOC Manuals and Guides No. 33. UNESCO, Paris.
- He, K., Zhang, X., Ren, S., and Jian, S. 2016. Deep residual learning for image recognition. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, pp. 770–778.
- He, K., Zhang, X., Ren, S., and Sun, J. 2015. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37: 1904–1916.
- Hosang, J., Benenson, R., and Schiele, B. 2017. Learning non-maximum suppression. *In Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4507–4515.
- Isola, P., Zhu, J. Y., Zhou, T., and Efros, A. A. 2016. Image-to-image translation with conditional adversarial networks. *In Proceeding of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1125–1134.
- Johnson, J., Alahi, A., and Li, F. F. 2016. Perceptual losses for real-time style transfer and super-resolution. *In European Conference on Computer Vision*, pp. 694–711.
- Jonker, R. R., Meulemans, J. T., Dubelaar, G. B. J., Wilkins, M. F., and Ringelberg, J. 1995. Flow cytometry: a powerful tool in analysis of biomass distributions in phytoplankton. *Water Science & Technology*, 32: 177–182.
- Kim, T., Cha, M., Kim, H., Lee, J., and Kim, J. 2017. Learning to discover cross-domain relations with generative adversarial networks. *In Proceedings of the 34th International Conference on Machine Learning*, pp. 1857–1865.
- Kolber, Z., and Falkowski, P. G. 1993. Use of active fluorescence to estimate phytoplankton photosynthesis in situ. *Limnology and Oceanography*, 38: 1646–1665.
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. 2012. ImageNet classification with deep convolutional neural networks. *In Advances in Neural Information Processing Systems*, Lake Tahoe, pp. 1907–1105.
- Lecun, Y., Bengio, Y., and Hinton, G. 2015. Deep learning. *Nature*, 521: 436.
- Ledig, C., Wang, Z., Shi, W., Theis, L., Huszar, F., Caballero, J., Cunningham, A. *et al.* 2017. Photo-realistic single image super-resolution using a generative adversarial network. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 105–114.

- Li, X., and Cui, Z. 2016. Deep residual networks for plankton classification. *Oceans 2016 MTS/IEEE Monterey*, pp. 1–4.
- Lin, T. Y., Dollar, P., Girshick, R., He, K., Hariharan, B., and Belongie, S. 2017a. Feature pyramid networks for object detection. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Hawaii*, pp. 2117–2125.
- Lin, T. Y., Goyal, P., Girshick, R., He, K., and Dollar, P. 2017b. Focal loss for dense object detection. *In IEEE Transactions on Pattern Analysis & Machine Intelligence*, pp. 2999–3007.
- Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., and Berg, A. C. 2016. SSD: single shot multibox detector. *In European Conference on Computer Vision, Amsterdam*, pp. 21–37.
- Long, M., Wang, J., and Jordan, M. I. 2017. Deep transfer learning with joint adaptation networks. *In Proceedings of the 34th International Conference on Machine Learning*, pp. 2208–2217.
- Lopez, M. M., and Kalita, J. 2017. Deep learning applied to NLP. arXiv preprint arXiv:1703.03091.
- Noda, K., Yamaguchi, Y., Nakadai, K., Okuno, H. G., and Ogata, T. 2015. Audio-visual speech recognition using deep learning. *Applied Intelligence*, 42: 722–737.
- Nohe, A., Knockaert, C., Goffin, A., Dewitte, E., De Cauwer, K., Desmit, X., Vyverman, W. *et al.* 2018. Marine phytoplankton community composition data from the Belgian part of the North Sea. *Scientific Data*, 5: 1968–2010.
- O'Brien, T. 2005. Copepod: a global plankton database. NOAA Technical Memorandum NMFS-F/SPO-73: 19–20.
- Olson, R. J., and Sosik, H. M. 2007. A submersible imaging-in-flow instrument to analyze nano- and microplankton: imaging FlowCytobot. *Limnology & Oceanography Methods*, 5: 195–203.
- Orenstein, E. C., Beijbom, O., Peacock, E. E., and Sosik, H. M. 2015. WHOI-Plankton—a large scale fine grained visual recognition benchmark dataset for plankton classification. arXiv preprint arXiv:1510.00745.
- Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., and Wang, B. 2018. Moment matching for multi-source domain adaptation. arXiv preprint arXiv:1812.01754.
- Poulton, N. J. 2016. FlowCam: quantification and classification of phytoplankton by imaging flow cytometry. *Methods in Molecular Biology*, 1389: 237.
- Rawat, W., and Wang, Z. 2017. Deep convolutional neural networks for image classification: a comprehensive review. *Neural Computation*, 29: 2352.
- Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. 2016. You only look once: unified, real-time object detection. *In Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pp. 779–788.
- Redmon, J., and Farhadi, A. 2017. YOLO9000: better, faster, stronger. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Hawaii*, pp. 7263–7271.
- Redmon, J., and Farhadi, A. 2018. YOLOv3: an incremental improvement. arXiv preprint arXiv:1804.02767.
- Ren, S., He, K., Girshick, R., Sun, J. 2015. Faster R-CNN: towards real-time object detection with region proposal networks. *In International Conference on Neural Information Processing Systems*, pp. 91–99.
- Schmidhuber, J. 2015. Deep learning in neural networks: an overview. *Neural Network*, 61: 85–117.
- Shelhamer, E., Long, J., and Darrell, T. 2014. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 39: 640–651.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15: 1929–1958.
- Sullivan-Silva, K. B., and Forbes, M. J. 1992. Behavioral study of zooplankton response to high-frequency acoustics. *Journal of the Acoustical Society of America*, 92: 2423–2423.
- Sun, X., Liu, L., Li, Q., Dong, J., Lima, E., and Yin, R. 2019. Deep pixel-to-pixel network for underwater image enhancement and restoration. *IET Image Processing*, 13: 469–474.
- Sun, X., Shi, J., Liu, L., Dong, J., Plant, C., Wang, X., and Zhou, H. 2018. Transferring deep knowledge for object recognition in low-quality underwater videos. *Neurocomputing*, 275: 897–908.
- Tzatalin, D. 2015. LabelImg. Git code. <https://github.com/tzatalin/labelimg> (last accessed 28 August 2019).
- Uijlings, J. R. R., Sande, K. E. A. V. D., Gevers, T., and Smeulders, A. W. M. 2013. Selective search for object recognition. *International Journal of Computer Vision*, 104: 154–171.
- Wang, J., He, Z., Feng, C., Zhu, Z., and Xie, S. 2018. Domain confusion with self-ensembling for unsupervised adaptation. arXiv preprint arXiv:1810.04472.
- Warren, J. D., Stanton, T. K., Benfield, M. C., Wiebe, P. H., Chu, D., and Sutor, M. 2001. In situ measurements of acoustic target strengths of gas-bearing siphonophores. *ICES Journal of Marine Science*, 58: 740–749.
- Watanabe, S., Hori, T., and Hershey, J. R. 2018. Language independent end-to-end architecture for joint language identification and speech recognition. *In IEEE Automatic Speech Recognition and Understanding Workshop, Okinawa*, pp. 265–271.
- Yan, Z., Hao, Z., Piramuthu, R., Jagadeesh, V., Decoste, D., and Wei, D., Yu, Y. 2015. HD-CNN: hierarchical deep convolutional neural networks for large scale visual recognition. *In Proceedings of the IEEE International Conference on Computer Vision, Santiago*, pp. 2740–2748.
- Yi, Z., Hao, Z., Ping, T., and Gong, M. 2017. DualGAN: unsupervised dual learning for image-to-image translation. *In Proceedings of the IEEE International Conference on Computer Vision, Venice*, pp. 2849–2857.
- Young, T., Hazarika, D., Poria, S., and Cambria, E. 2018. Recent trends in deep learning based natural language processing. *IEEE Computational Intelligence Magazine*, 13: 55–75.
- Yu, L., Song, J., and Tang, H. 2004. *In situ* capturing of plankton images. *Ocean Technology of China*, 25: 47–49.
- Yu, Z., Chan, W., and Jaitly, N. 2017. Very deep convolutional networks for end-to-end speech recognition. *In IEEE International Conference on Acoustics, Speech and Signal Processing, New Orleans*, pp. 4845–4849.
- Zhu, J. Y., Park, T., Isola, P., and Efros, A. A. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. *In Proceedings of the IEEE International Conference on Computer Vision, Venice*, pp. 2223–2232.
- Zitnick, C. L., and Dollár, P. 2014. Edge boxes: locating object proposals from edges. *In European Conference on Computer Vision*, pp. 391–405.

Handling editor: Cigdem Beyan



Contribution to the Themed Section: 'Applications of machine learning and artificial intelligence in marine science'

Original Article

The Prince William Sound Plankton Camera: a profiling *in situ* observatory of plankton and particulates

R. W. Campbell ^{1*}, P. L. Roberts², and J. Jaffe³

¹Prince William Sound Science Center, Box 705, Cordova, AK 99574, USA

²Monterey Bay Aquarium Research Institute, 7700 Sandholdt Road, Moss Landing, CA 95039, USA

³University of California San Diego, 9500 Gilman Dr, M/C 0238, La Jolla, CA 92093-0238, USA

*Corresponding author: tel: +1 907 424 5800; e-mail: rcampbell@pwssc.org.

Campbell, R. W., Roberts, P. L., and Jaffe, J. The Prince William Sound Plankton Camera: a profiling *in situ* observatory of plankton and particulates. – ICES Journal of Marine Science, 77: 1440–1455.

Received 9 August 2019; revised 13 December 2019; accepted 11 February 2020; advance access publication 24 March 2020.

A novel plankton imager was developed and deployed aboard a profiling mooring in Prince William Sound in 2016–2018. The imager consisted of a 12-MP camera and a 0.137× telecentric lens, along with darkfield illumination produced by an in-line ring/condenser lens system. Just under 2.5×10^6 images were collected during 3 years of deployments. A subset of almost 2×10^4 images was manually identified into 43 unique classes, and a hybrid convolutional neural network classifier was developed and trained to identify the images. Classification accuracy varied among the different classes, and applying thresholds to the output of the neural network (interpretable as probabilities or classifier confidence), improved classification accuracy in non-ambiguous groups to between 80% and 100%.

Keywords: convolutional neural network, machine vision, Prince William Sound, zooplankton.

Introduction

There is a considerable interest in moving marine resource management away from a single-species approach to a more mechanistic ecosystem approach (e.g. Friedland *et al.*, 2012), but that has proved to be challenging in practice given the complexity and variability of large marine ecosystems. Technologies for measuring physical parameters (temperature and salinity) are mature, and technologies for measuring biogeochemical parameters (e.g. nitrate, phosphate) are also now available and reasonably robust (Johnson *et al.*, 2006). Similarly, the practice of using *in situ* fluorescence as a proxy for primary producer biomass is well established (e.g. Strickland and Parsons, 1972). The assessment of higher trophic levels, particularly fish stocks, is also mature and features an array of well-developed methods (e.g. King, 2007). Zooplankton are the link between primary productivity and fisheries, but zooplankton studies have often been sidelined within

ecosystem studies (Mitra *et al.*, 2014) because they are difficult and expensive to enumerate.

In high latitude ecosystems, secondary producers are mostly small Eumetazoan zooplankton (Longhurst, 2006). Although the dominant large grazers are often crustaceans, there is a diversity of other taxa present: most every phylum within the subkingdom has a member that may be found in the plankton during at least part of their life history. Until recently, the assessment of zooplankton was primarily done by collecting them with nets and examining the resulting samples under a microscope. This method is time consuming and expensive and destroys fragile taxa but is required if species-level taxonomic resolution is desired. There has been much work in recent years on new methods to enumerate zooplankton taxa, both *in situ* and *in manus* (reviewed by Wiebe and Benfield, 2003). One of the more promising methods has proved to be *in situ* imagery, which permits the

discrimination of plankton from abiotic particulates, provides a level of taxonomic resolution, and sizing of imaged plankton and particulates (which is useful because biomass, and many physiological rates scale with size).

A number of *in situ* imagers have been developed, including traditional camera-based systems such as the Video Plankton Recorder (Davis *et al.*, 1992), ZOOVIS (Benfield *et al.*, 2003), and the Scripps Plankton Camera (spc.ucsd.edu). A number of systems have also been developed that employ shadowgraph imagery (Samson *et al.*, 2001; Cowen and Guigand, 2008; Ohman *et al.*, 2019); shadowgraph systems possess a very long depth of field and consequently permit sampling large volumes of water. The disadvantage of shadowgraph systems is that only the silhouette of non-transparent plankton is recovered and only greyscale images may be collected.

Given the high abundance of zooplankton *in situ* (order 10^2 – 10^6 individuals l^{-1}), most imaging systems collect many more images than may be identified manually and there has also been a parallel effort to develop machine vision techniques to automate the identification of those images (Benfield *et al.*, 2007). Early methods included discriminant analysis (Jeffries *et al.*, 1984), and more recently Support Vector Machines and Artificial Neural Networks (e.g. Culverhouse *et al.*, 1996; Hu and Davis, 2005) and Random Forest (Gorsky *et al.*, 2010) methods have been employed successfully.

With recent advances in computing hardware, most notably the development of cost-effective massively parallel graphics processing unit (GPU) based processors, very deep convolutional neural networks (CNNs) have been developed for solving complex computer vision problems such as image classification (Krizhevsky *et al.*, 2012). CNNs of varying architecture are now commonly employed to address the classification of plankton images from *in situ* imaging systems (e.g. Cui *et al.*, 2018; Luo *et al.*, 2018; Schröder *et al.*, 2018; Bochinski *et al.*, 2019; Cheng *et al.*, 2019). Many studies have focused on smaller phyto- and microzooplankton images based on the publicly available WHOI database (Orenstein *et al.*, 2015; Sosik *et al.*, 2015) and report accuracies in the range of 86–96% (e.g. Lee *et al.*, 2016; Cui *et al.*, 2018; Liu *et al.*, 2018). Among larger zooplankton, Luo *et al.* (2018) used a CNN to identify shadowgraph images to a classification accuracy of order of 90%, if rare difficult-to-classify groups were omitted. Bochinski *et al.* (2019), using a similar image set, reported accuracies between 69% and 98%. Cheng *et al.* (2019) showed accuracies of between 91% and 98% on a seven-class set of shadowgraph images collected by the ZOOVIS camera (Bi *et al.*, 2013). Transfer learning, the use of pre-trained very deep CNNs has been shown to improve both speed and accuracy when classifying plankton image sets (Lee *et al.*, 2016; Orenstein and Beijbom, 2017; Rodrigues *et al.*, 2018; Schröder *et al.*, 2018).

As a part of the GulfWatch Alaska programme (gulfwatch.com), a long-term monitoring effort in the area impacted by the Exxon Valdez oil spill, a WETLabs Autonomous Moored Profiler (AMP) has been deployed in central Prince William Sound annually since 2013. The AMP site is ~5 nautical miles southeast of Naked Island, in 200 m water depth. The AMP system is a surface piercing profiler that profiles from a parking depth to surface at a user-specified rate and interval. Once at the surface, the profiler connects to a server computer on land via a cellular data link for data upload and command/control telemetry and then pulls itself back down the line to the park depth with a small onboard winch.

In 2015, an *in situ* zooplankton camera system was developed for the PWS AMP. The camera system was based on the Scripps Plankton Camera, but with larger optics and a higher resolution camera, to sample a larger volume of water to better sample mesozooplankton. The camera system was integrated with the profiler electronics and deployed on the profiler during deployments in 2016–2018. We present here a description of the camera system and a CNN-based classification system that was developed using the images collected during the deployments.

Methods

PWS profiler

The PWS AMP system is based on a WETLabs Thetis profiler, which consists of a positively buoyant frame (~20 lbs), an electric winch, and a 2.8-mm UHMWPE tether. Starting from a user-specified parking depth, the winch pays out the tether at a specified rate to allow the profiler to ascend. Upon reaching the surface, the profiler enters into a “hold” mode, while an onboard cellular modem connects to the local cellular network. Upon connecting, new profile parameters may be sent to the profiler and a small amount of decimated data from the profile sent out. Following that, or if the profiler is unable to connect to the cellular network before a timeout period (as will occur during heavy weather), it engages the winch and pulls the frame back down to the park depth (Figure 1). The system is powered by a 1.5-kW lithium polymer battery manufactured by Bluefin Robotics for autonomous underwater vehicle use, and with the current configuration it is capable of conducting ~70 60-m profiles per charge.

The instrument suite on the AMP includes a Seabird model 19 CTD, a WETLabs FLNTU chlorophyll-*a* fluorometer/backscatter turbidometer, a Satlantic SUNA nitrate sensor, and a Seabird SBE43 oxygen sensor. During the 2016–2018 deployments, the profiler was set to conduct twice daily profiles from 60 m depth to the surface. Profiles were usually done within 15 min of the solar minimum and maximum of each day. The ascent rate was set to 30 cm s^{-1} .

PWS Plankton Camera

The optical system of the PWS Plankton Camera (PWSPC) includes a $0.137\times$ 143-mm telecentric lens (Opto Engineering TC2MHR-96) mounted on a 12-MP colour camera (a Point Grey Grasshopper GS3-U3-120S6C-C) inside a large pressure housing with a sapphire glass optical port (Figure 2). Illumination is provided from a second pressure housing on titanium standoffs aimed at the imaging system, with a custom white light emitting diode (LED) array focused through condenser lenses (Edmund Optics 125 mm plano-convex anti-reflective coated lenses) and a white LED ring ahead of the condenser lenses, to produce darkfield illumination of the imaged volume (Figure 2). The LEDs are strobed with a control signal from the camera to synchronize with the frame rate. The imaged volume of the camera is ~450 ml, and the nominal pixel size is 22.6 μm .

The camera takes 12-bit colour images at a maximum frame rate of 7 frames s^{-1} , which produces more data that can be practically logged to disk (~500 MB s^{-1} , or ~5.5 TB for a 1 month deployment of twice daily profiles lasting 3 min). However, mesozooplankton are sparse enough that most of each frame does not contain an image of a particle, it is mostly empty space. The PWSPC thus also incorporates an onboard computer (an Odroid XU4) to segment each image and retain regions of

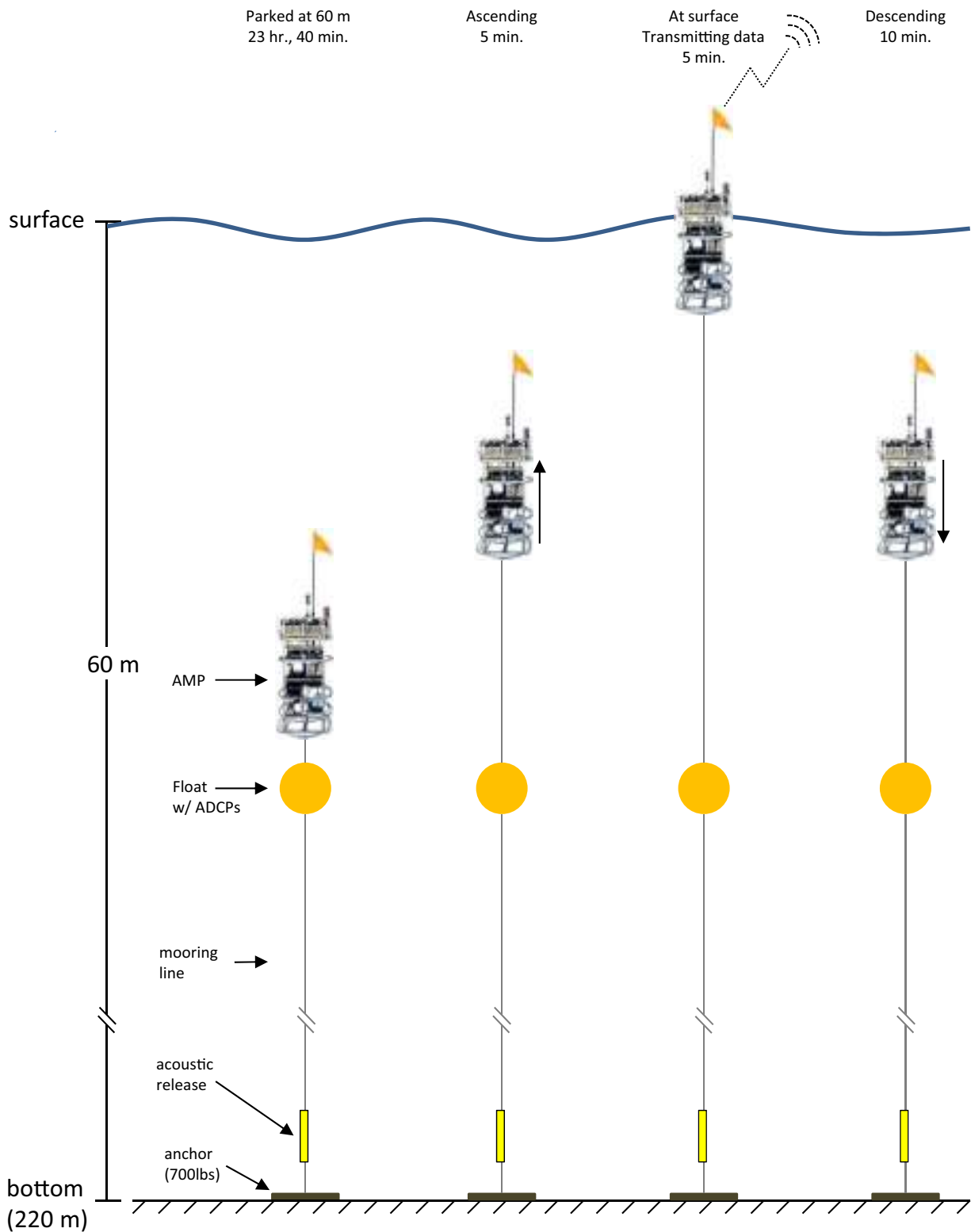


Figure 1. Schematic representation of the PWS profiling mooring and its operation.

interests (ROIs) that contain images of individual plankters. Raw input images were downsampled by a factor of 4 using nearest-neighbour interpolation and then scaled to 8 bits by dividing the pixel values by 256. This approach preserves

resolution in one colour channel and avoids the computationally costly debayering operation on the full 12-MP image. ROIs in each frame were detected with the Canny algorithm (Canny, 1986), a multi-step algorithm commonly used to detect edges in

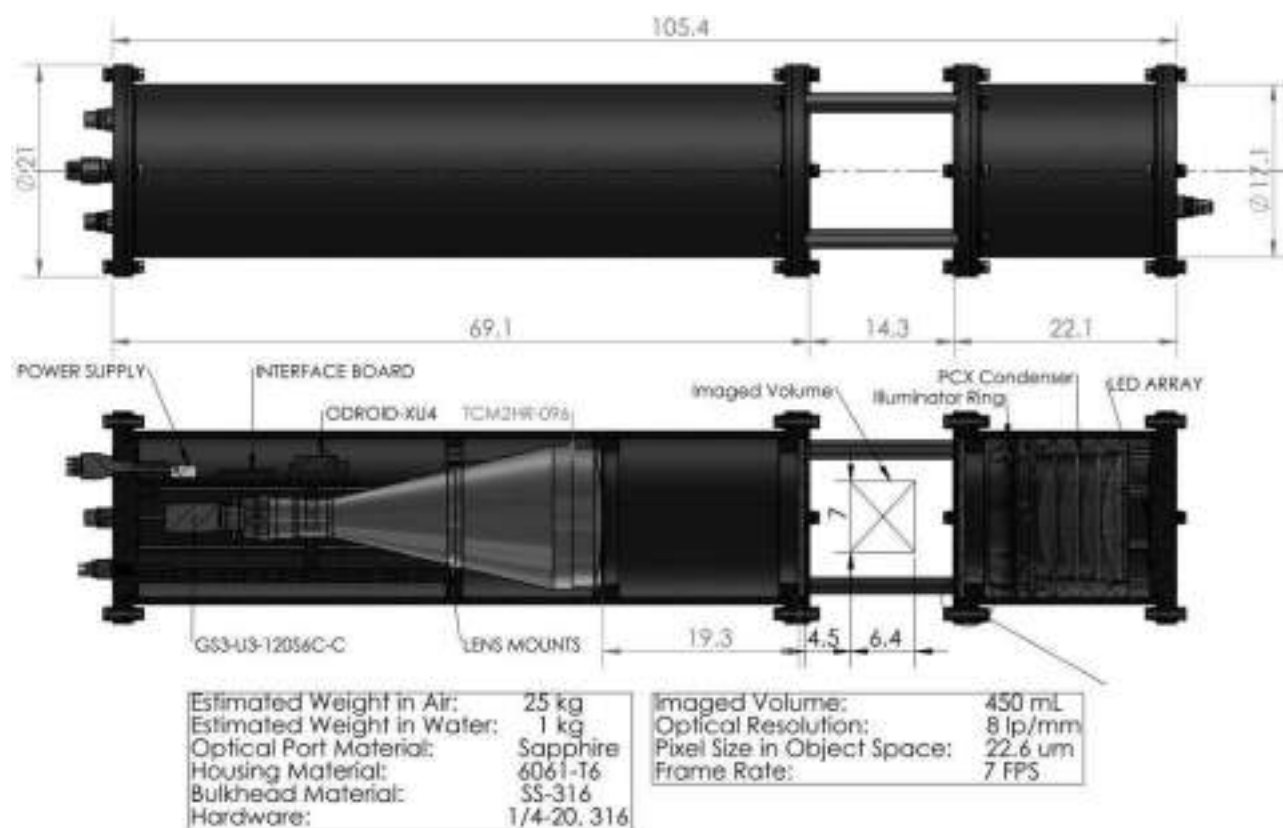


Figure 2. Schematic representation of the PWS Plankton Camera.

images. High and low thresholds were set at 50 and 100, respectively, and kernel size was 3; those thresholds were empirically set and gave good detection of objects in sharp focus or with very high contrast. The edge mask from the Canny operation was then post-processed with a binary morphological closing operation (dilation followed by erosion) with a 5×5 kernel to bridge disjoint edges together. Contours were then detected in the image using the OpenCV findContours function. For each contour with area larger than a threshold of 300 pixels, the contour bounding box was padded by a factor of 50% and upsampled to the scale of the raw image. The coordinates of the padded and upsampled bounding box were then used to extract the ROI from the raw input image and save it to disk in raw 16-bit TIFF format. An upper limit of 50 ROIs per image was imposed by hardware limitations and was not often reached: over the 3 years of deployments, the mean number of ROIs per image was 7.8, and the 50 ROI limit was reached only 0.36% of the time. Images were downloaded from the camera over Gigabit Ethernet during regular service visits to the profiler done every 4–6 weeks.

The PWSPC was integrated with the AMP electronics, and control of the camera system done via an RS232 serial link. Prior to each profile, the AMP control module supplied power to the camera and waited for the onboard computer to boot. After the computer had booted, the AMP sent a string to synchronize the computer clock and an instruction to start logging and then started the profile. As the profile occurred, the PWSPC computer output status messages (time, number of ROIs collected, status messages from the various components) at 1 Hz that were logged by the AMP electronics. ROIs saved to the onboard disk were

Table 1. Specifications of the PWS Plankton Camera.

Exposure time (μ s)	10–60
Magnification	0.137 \times
Field of view (mm)	93 \times 70
Pixel size (object space) (μ m)	22.6
Optical resolution	8 lp/mm at 30% contrast
Depth of field	64 mm at 8 lp/mm at 20% contrast
Full-resolution imaged volume (ml)	400
Blob detection imaged volume (ml)	> 1 000
Frame rate	4 frames/s with ROI processing
Onboard storage (GB)	64
Dimensions (excluding cables)	120 cm L \times 18 cm OD
Total system weight (kg)	\sim 10 (air), \sim 2 (seawater)
Power requirements	9–36 V input, 20 W consumption
External communications	RS232, 100 Mbit Ethernet

given timestamped filenames to be used to infer the depth of the profiler at the time each image was taken from the pressure record recorded by the CTD. Following the profile, the AMP controller shut down the computer and removed power to the camera system before returning to the park depth. During profiling the camera and strobes were set to operate at 4 Hz to prevent overlapping images from being taken. The technical specifications of the PWSPC are outlined in Table 1.

Image preprocessing and CNN classifier

Prior to analysis, 16-bit ROIs were debayered to produce a colour image at full camera resolution. These colour images were then

contrast-enhanced by subtracting the minimum and dividing by the maximum of the image. The contrast-enhanced images were then converted to 8 bits by multiplying by 255 and coercing to integer values. The full-resolution colour images were then post-processed using a method similar to the real-time detection method. The images were first converted to greyscale and then filtered with a Sobel edge detector. The edge magnitude image was then thresholded by setting edge magnitudes >2.5 times the median edge magnitude to 255 and others to 0. The edge image was then closed using binary morphological operations, and closed contours are enumerated. The contour with the largest area was then selected as the foreground object. Finally, the contour mask was smoothed with a Gaussian filter and the mask multiplied with the colour image. Each colour channel of the resulting masked image was then deconvolved with the Lucy–Richardson algorithm with seven iterations and a Gaussian point spread function estimate with full width half maximum set to three pixels. The deconvolved colour channels were then combined together to yield the masked, sharpened, colour ROI.

The CNN chosen to classify the PWSPC images was the “Inception v3” model (Szegedy et al., 2015). Inception v3 is a very deep CNN with numerous symmetric and asymmetric neurons that has proved to be adept at image classification problems, and it and its predecessors have consistently ranked highly in the ImageNet Large Scale Recognition Competition (Russakovsky et al., 2015). The ImageNet database for the 2015 competition included ~ 1.2 million images in 1000 unique categories; the Inception v3 model had an overall error rate of 5.6%. Because the model is very large, it can be very time intensive to train from scratch, but pre-trained weights (such as from the ImageNet competition) may be used to significantly reduce training time (Tajbakhsh et al., 2016; Orenstein and Beijbom, 2017). To work with the Inception v3 model, the shorter dimension of each image was padded with black values (red-green-blue 0, 0, 0) to make a square image. Both sides of the image were padded to approximately centre the image. Images were rescaled to a dimension of 299×299 pixels (the default size of the model).

Resizing the images for input to the CNN necessarily discards size information that is encoded in the image, which can lead to confusion among similar looking but differently sized plankton. For instance, *Pseudocalanus* and *Neocalanus* copepods have a similar appearance but are fairly easily distinguished by size. To reintroduce size information to be used to improve classification, a hybrid architecture was employed, with a second parallel neural network developed to operate on a small set of features extracted from each image, including the major and minor axis lengths and areas in pixels. The first 12 Haralick texture features (Haralick, 1979), a common set of statistics used for image classification (Hu and Davis, 2005), were also included. The features were encoded into a single neuron, batch-normalized, and concatenated with the Inception v3 model prior to the last two layers to produce a hybrid model (Figure 3). The network was implemented in Tensorflow (Abadi et al., 2016) through the Keras front end (Chollet, 2015) in the Python programming language. Training of the network and image classification was done on an NVIDIA Tesla K40 GPU.

CNN training

To produce a training set, images were randomly subsampled from the entire image set. Because the size frequency distribution

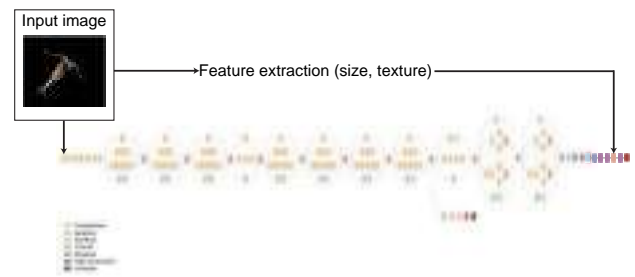


Figure 3. Schematic representation of the Inception v3 CNN and concatenated feature size and texture model.

of the images was roughly lognormal (Figure 4), sampling randomly from the entire set produced batches of images that were mostly smaller particles, which also tended to be of lower resolution and more difficult to identify. Therefore, the images were stratified into four logarithmically scaled size groups based on file size (<1642 ; $>1642, \leq 10\,000$; $>10\,000, \leq 28\,183$, and $>28\,183$ bytes) before being identified. File size is a useful proxy of image size, and this subsampling scheme allowed more larger images to be classified, which were more likely to be identifiable mesozooplankton. The image set was further stratified by time, such that approximately one-third of the images were taken from each of the 3 years, to provide a subsample representative of all the images.

The amount of training data available is a bottleneck in the training process of CNNs; for complex classification tasks, a large training dataset (10^5 images or more) is desirable. To streamline the identification of the stratified subsets, a custom programme was developed in the Matlab GUIDE framework. The programme consists of a graphical interface that presents an observer with the image displayed at its actual size alongside a larger zoomed version and has a text box into which descriptive text may be entered. Upon entering text and pressing enter, the identification is recorded and the next image in the set presented. Using the GUI, each image could be identified in a few seconds, allowing a large number of images to be identified in a relatively short time.

The training set was produced by an expert zooplankton taxonomist, and each image was identified to the finest taxonomic resolution possible. The training set produced contained 18 868 images within 43 separate classes; some classes were taxon based, while others were based on visual characteristics (Figure 5, Table 2). A number of rare classes (<10 images) were identified during manual classification but were not included for analysis.

The model was initialized with ImageNet weights and was trained using categorical crossentropy as the loss function and the Adam optimizer (Kingma and Ba, 2015); accuracy was the primary metric. The training set was split randomly 90/10 into a training and test set, and 10% of the training set was used for validation purposes during training. Image augmentation (Perez and Wang, 2017) has been shown to improve classification accuracy in classification problems with relatively small amounts of training data and was applied to the images during training. Images were randomly flipped, scaled ($\pm 20\%$), rotated ($\pm 90^\circ$), or sheared ($\pm 8^\circ$) as they were input into the model during each training epoch. Network parameters were only retained if they resulted in an increase in validation accuracy.

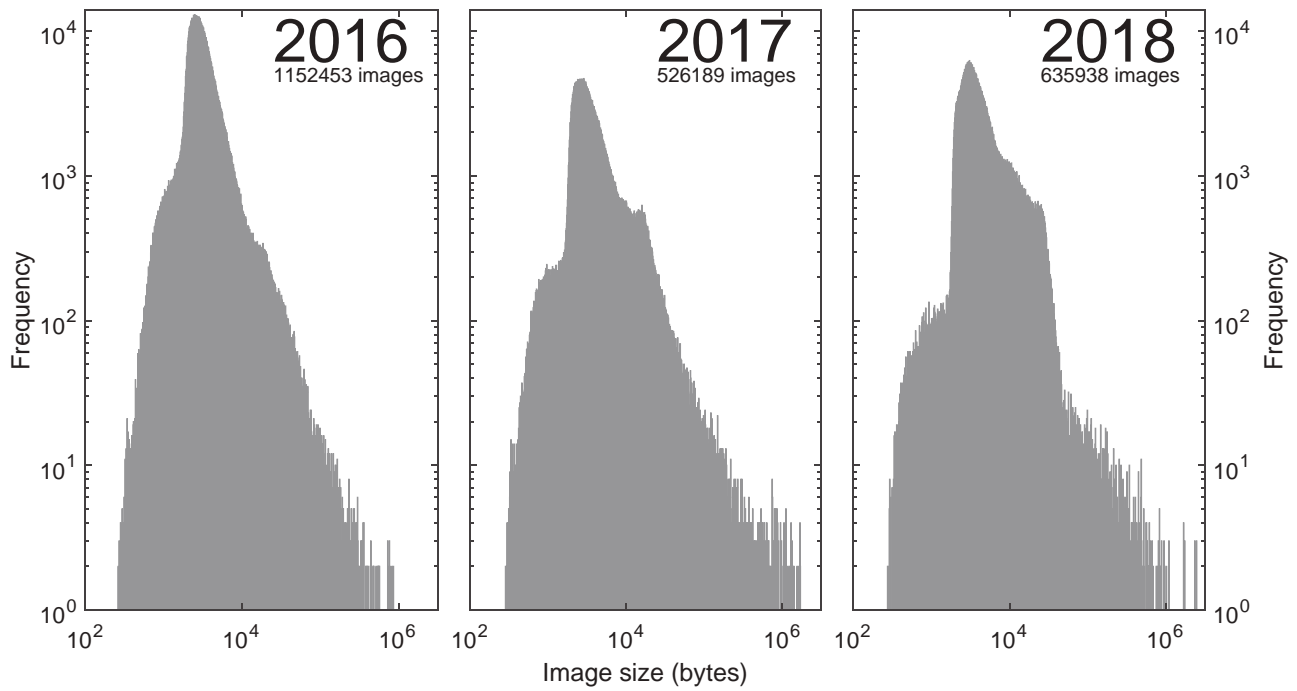


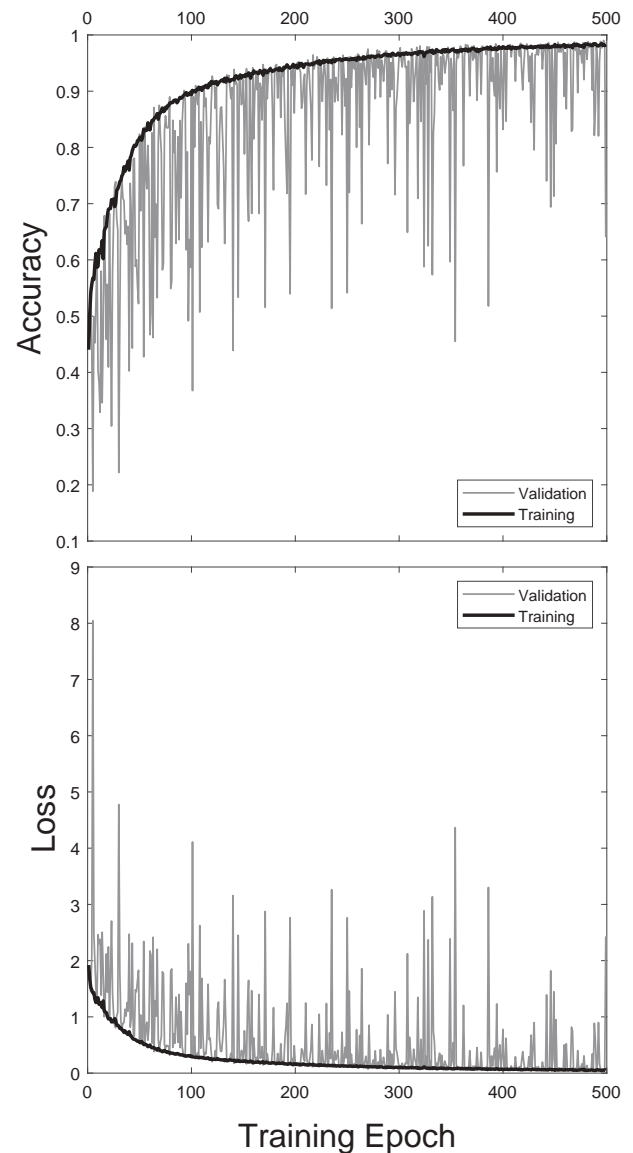
Figure 4. Size frequency histograms of ROI sizes during the 3 years of deployments.



Figure 5. Examples of non-ambiguous taxa groups among the 43 unique classes identified. Scaling is consistent among different taxa, and the number corresponds to the taxa group number in [Table 2](#).

Table 2. Description of the 43 classes in the training set.

Name	Number		Notes
	Number	of images	
Acartia	1	100	
Aegina	2	147	
Aglantha	3	330	
Amphipoda	4	100	Merged several amphipod species to produce larger group
Beroe	5	100	
Blob	6	2 989	Characteristic opaque large single cell. Possibly Noctiluca
Bolinopsis	7	255	
Calanus	8	613	
Calyptopis	9	198	
Chaetognatha	10	212	
Clione	11	100	
Clytia	12	100	
Cnidaria	13	262	Catchall group of several uncommon species and images not identifiable to species
Cope_lg	14	815	Catchall group of large copepods (approximately Calanus/Metridia sized and larger) not identifiable to species
Cope_sm	15	1 117	Catchall group of large copepods (approximately Pseudocalanus sized and smaller) not identifiable to species
Ctenophora	16	174	Catchall group of Ctenophora not identifiable to lobate groups or Pleurobrachia
Doliolida	17	103	
Dot	18	110	Image artefact: small white dots
Eucalanus	19	177	
Euphausiid	20	97	Juvenile and larger
Filament	21	656	Long thin forms likely diatom chains or large pennate diatoms.
Filaments	22	199	Multiple filaments, often poorly segmented cnidarian tentacles
Furcilia	23	114	Euphausiid furcilia
Larvacea	24	100	Catchall group for non-identifiable and not Oikopleura
Limacina	25	210	
Metridia	26	1 353	
Nauplius	27	256	Nauplii of all types, taxonomically ambiguous
Neocalanus	28	1 574	
Oikopleura	29	207	House usually segmented out
Oithona	30	197	
Paraeuchaeta	31	100	
Pleurobrachia	32	262	
Pluteus	33	308	Primarily echinoderm pluteus larvae
Polychaeta	34	100	Catchall for all polychaetes not identifiable as Spionidae
Pseudocalanus	35	1 004	
Radiolarian	36	251	
Siphonophora	37	204	
Snow	38	172	Amorphous aggregates
Spionidae	39	99	
Spiral	40	177	
Tentacle	41	132	Cnidarian tentacles
Tentacles	42	145	Multiple tentacles in frame.
Unknown	43	2 949	

**Figure 6.** Training and validation accuracy (top panel) and loss (bottom panel) over the 500 training epochs.

Results

A total of 2 424 329 ROIs totalling just over 60 GB were collected during the 2016–2018 deployments (Figure 4).

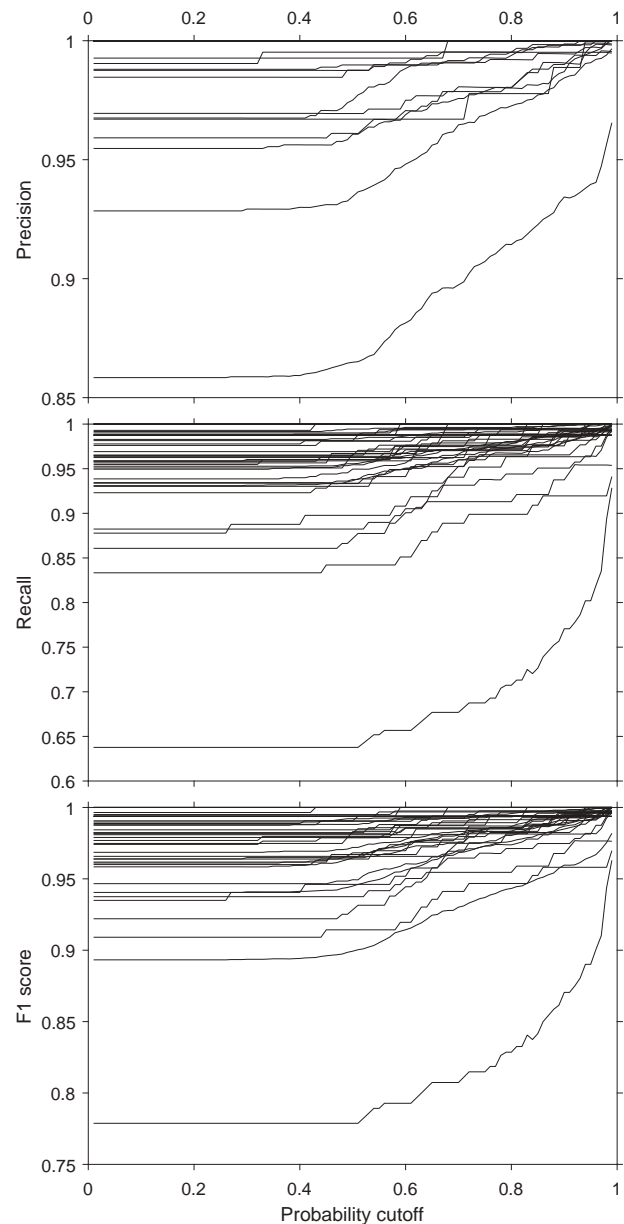
The Inception v3 model was trained on the training set for 500 epochs and took ~ 275 s per epoch, taking slightly under 40 h. Training accuracy increased to $>90\%$ by the 100th epoch, and the rate of increase in training accuracy declined after that, slightly exceeding 98% by the 500th epoch (Figure 6). Validation accuracy and loss was much more variable, presumably due to variability in the image set from epoch to epoch but followed the same trend.

A confusion matrix is a method of representing the accuracy of the classifier (Pearson, 1904; Hu and Davis, 2005; Luo *et al.*, 2018); the confusion matrix of the classifier run on the training data is the theoretical maximum performance that can be expected from the classifier (Figure 7). Furthermore, there are a

Table 3. Accuracy metrics of the training set (left) and test set (right) for each class.

Class name	Training set			Test set		
	Precision	Recall	F1 score	Precision	Recall	F1 score
Acartia	1	1	1	0.7	0.88	0.78
Aegina	1	1	1	0.87	1	0.93
Aglantha	1	0.99	1	0.91	0.83	0.87
Amphipoda	1	1	1	0.95	0.86	0.9
Beroe	1	1	1	1	0.83	0.91
Blob	0.98	0.96	0.97	0.85	0.88	0.86
Bolinopsis	1	1	1	0.88	0.76	0.82
Calanus	0.99	0.98	0.99	0.52	0.53	0.52
Calyptopis	1	1	1	0.78	0.94	0.85
Chaetognatha	1	1	1	0.84	0.82	0.83
Clione	1	1	1	1	1	1
Clytia	1	1	1	0.65	0.62	0.63
Cnidaria	1	0.97	0.99	0.25	0.35	0.29
Cope_lg	1	0.98	0.99	0.47	0.54	0.5
Cope_sm	0.99	0.96	0.98	0.51	0.53	0.52
Ctenophora	1	0.99	1	0.14	0.26	0.19
Doliolida	1	1	1	0.62	0.87	0.72
Dot	0.99	0.73	0.84	0.77	0.53	0.63
Eucalanus	1	1	1	0.92	0.97	0.94
Euphausiid	1	0.96	0.98	1	1	1
Filament	1	0.97	0.98	0.79	0.57	0.66
Filaments	1	0.99	0.99	0.7	0.68	0.69
Furcilia	0.97	0.98	0.97	0.78	0.78	0.78
Larvacea	1	0.99	0.99	0.6	0.6	0.6
Limacina	1	0.98	0.99	0.74	0.74	0.74
Metridia	0.99	1	0.99	0.89	0.84	0.87
Nauplius	1	0.96	0.98	0.88	0.77	0.82
Neocalanus	0.99	1	0.99	0.72	0.81	0.77
Oikopleura	1	0.97	0.99	0.76	0.71	0.74
Oithona	1	0.98	0.99	0.6	0.55	0.57
Paraeuchaeta	1	1	1	0.9	0.9	0.9
Pleurobrachia	1	0.99	1	0.74	0.72	0.73
Pluteus	1	1	1	0.97	0.9	0.93
Polychaeta	1	0.93	0.96	0.8	0.43	0.56
Pseudocalanus	1	1	1	0.7	0.69	0.69
Radiolarian	1	1	1	0.94	0.98	0.96
Siphonophora	1	0.99	0.99	0.71	0.78	0.74
Snow	1	0.95	0.98	0.63	0.79	0.7
Spionidae	1	0.95	0.98	0.85	0.81	0.83
Spiral	1	0.96	0.98	0.92	0.57	0.7
Tentacle	0.99	0.99	0.99	0.89	0.71	0.79
Tentacles	1	1	1	0.86	0.93	0.89
Unknown	0.91	0.98	0.95	0.68	0.7	0.69

assessment of the usefulness of the classifier is to test it on a set of images that it did not see during training (the 10% of images set aside as a “test” set). Deep neural networks tend to overfit to the training set, and image augmentation and aggressive use of dropout layers used in the Inception v3 model are techniques to reduce that (Yamashita *et al.*, 2018). When the classifier was applied to the test set, there were considerably more confusion and lower scores in all of the accuracy metrics (Figure 8; Table 3). If the overall success of the classifier with the different taxa is summarized by sorting by the F1 score (Figure 9), some taxa were resolved quite well, while others were not. Less populated groups

**Figure 8.** Changes in the accuracy metrics (precision, recall, and F1 score) in all of the 43 different classes of the training set as a function of varying the probability cut-off.

were not less likely to be classified accurately, several of the smaller classes were classified well (many were visually distinctive), and several larger groups (which were more heterogeneous visually) had lower success.

The presence of “unknown” (cannot be identified by a human observer) and novel (not seen before by the network) categories is problematic for CNNs, since their structure assumes a fixed and known set of classes. The softmax function used as the final layer in the Inception model returns scaled outputs that sum to 1 and may be interpreted as probabilities (Bridle, 1990; Goodfellow *et al.*, 2016); the prediction made by the classifier is usually assigned to the category with the highest associated probability.

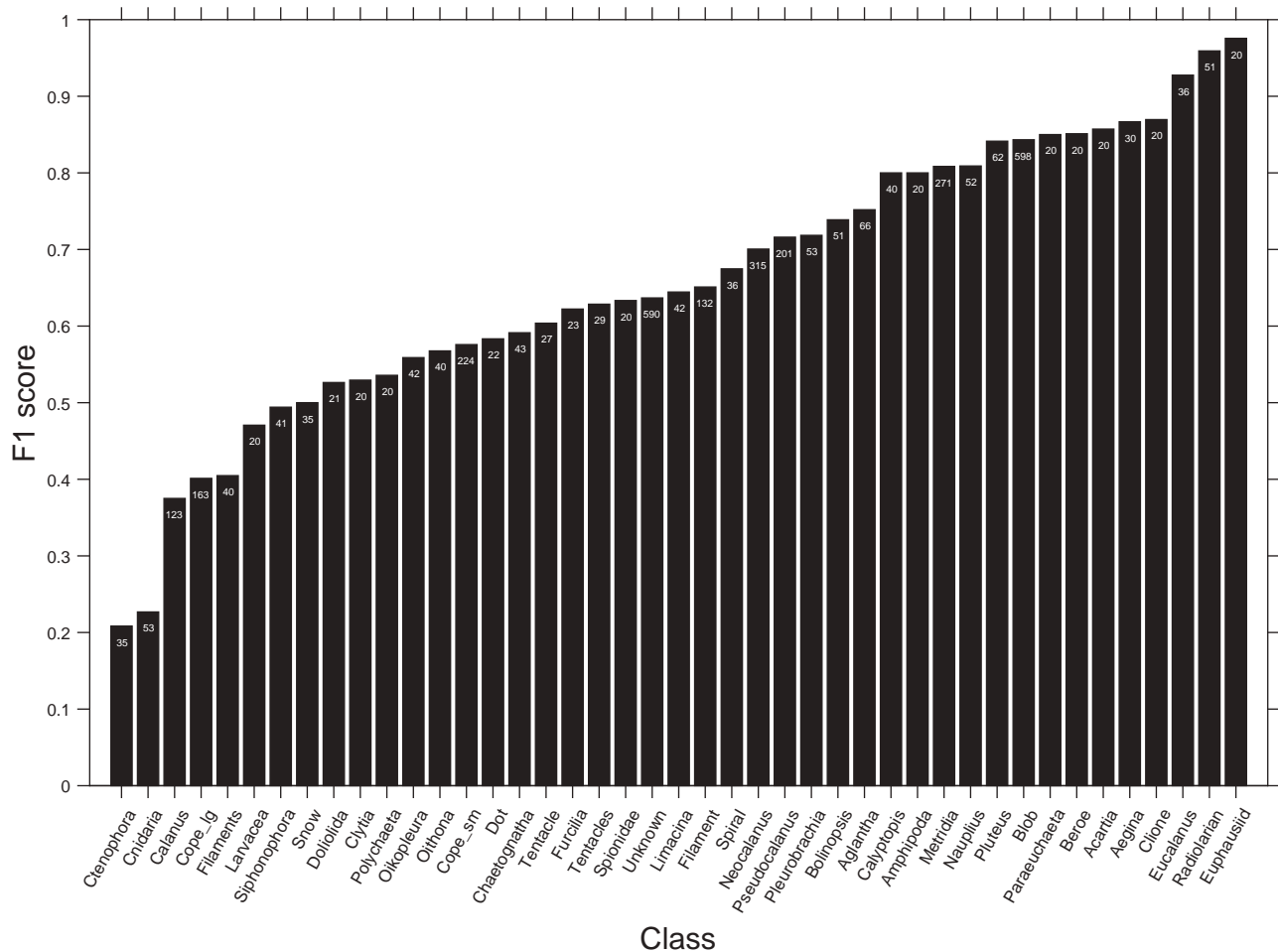


Figure 9. F1 scores for each of the 43 different classes in the test set.

The associated probability may also be used as a threshold to reduce the amount of misclassification and new groups that have not been encountered by the model before (Hendrycks and Gimpel, 2018). The technique has been used successfully with plankton images (Faillietaz *et al.*, 2016; Luo *et al.*, 2018).

To examine how a probability threshold might improve classification accuracy, the accuracy statistics were recalculated at varying probability thresholds (i.e. if the prediction for a particular image did not exceed the threshold it was not included in the calculation). Applying this procedure to all classes in the train and test sets (Figures 8 and 10, respectively) produced “trajectories” for each class that generally showed that a more restrictive probability threshold resulted in improvements in classification accuracy. The three taxa with the lowest F1 scores in the test set (Figure 9) showed an opposite trend, with a decrease in accuracy metrics at higher probability thresholds. Those classes were among the more ambiguous ones (“Ctenophora”, “Cnidaria”, “Calanus”) that exhibited high confusion with other classes with similar or even overlapping appearance (e.g. “Calanus” and “Cope-lg”; see Figure 11). Because those classes were employed when the human observer had low confidence of the identification, it is perhaps unsurprising that the confidence of the machine classifier remained low as well. A trade-off to this

technique is that as higher probabilities are used, more images are discarded from the analysis (Figure 12). If a 90% threshold is used, the overall error rate drops from ~30% to ~10%, but approximately a fifth of the images are discarded. A 95% threshold results in ~25% of images being discarded. Applying a 90% probability threshold resulted in an increase in most accuracy statistics in most classes (Figure 13, Table 4).

Discussion

The camera system developed here is among the highest resolution *in situ* zooplankton camera systems deployed thus far, with a comparatively large sampled volume as well (Table 5). It is also among the first colour imagers deployed, joining the Video Plankton Recorder (Davis *et al.*, 1992; Lombard *et al.*, 2019) and CPICS (Continuous Particle Imaging and Classification System: Grossmann *et al.*, 2015). Given that the system was designed for battery-limited autonomous vertical profiling (as opposed to long tows), a relatively large sampling volume was desirable, to capture adequate numbers of relatively dilute mesozooplankton (Sheldon and Parsons, 1967) during each profile. Colour information is also useful, because it may be diagnostic of some plankton classes (e.g. red pigments are common in some copepod

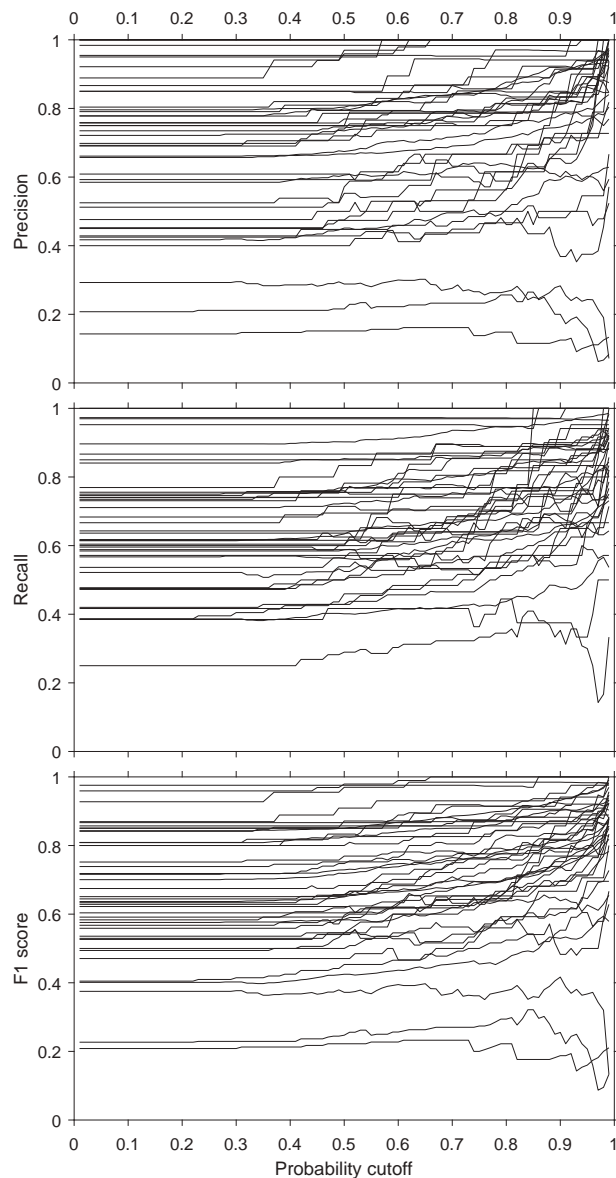


Figure 10. Changes in the accuracy metrics (precision, recall, and F1 score) in all of the 43 different classes in the test set as a function of varying the probability cut-off.

species), and features of the plankters are also discernable in some images (e.g. full guts and lipid sacs in copepods).

The image set collected during the 2016–2018 deployments spanned large phytoplankton to large mesozooplankton and exhibited a diversity of taxa, orientations, and qualities. Although magnification is constant across the depth of field with a telecentric lens, they do not have an infinite depth of field and particles on either side of the depth of field will be less sharp than those in the centre. Scattering by small particulates (phytoplankton cells too small to resolve and inorganic particles) may also have reduced the practical resolution at times. Not all images were sharp enough to detect the features required to identify a plankter to a fine taxonomic level. Larger raw images obviously had more features (*sensu* Hassaballah and Awad, 2016) that were more useful to a human observer and presumably to a machine observer as

well; smaller raw images, when upsized to 299×299 pixels for classification, remained less sharp.

Identification of taxa from images is a difficult task, and the error rate of human observers can be significant. In a dinoflagellate classification task, Culverhouse *et al.* (2003) found that expert taxonomists achieved 84–95% accuracy at best, although accuracy dropped considerably among multiple observers (43%). Similar studies in other fields have shown lower ranges in more visually complex situations (69–96%: Austen *et al.*, 2017). The proportion of unknown images can also vary among observers and can depend on the number of classes involved (Cowen *et al.*, 2015). Luo *et al.* (2018) suggest that 90% accuracy be used as a benchmark for automated classification. Those levels of accuracy were possible with the Inception v3 CNN for a number of taxa, particularly if images with lower confidence were not used.

Filtering images by probability, as suggested by Faillettaz *et al.* (2016), improved precision and recall in most taxa by 5–10%. Examination of the confusion matrix post filtering (Figure 13) shows that much of the confusion was between related classes, for instance the large calanoid copepods *Metridia*, *Calanus*, *Neocalanus*, and the catchall group Cope_lg and the small copepod classes *Oithona*, *Pseudocalanus*, and the catchall group Cope_sm. There was also confusion among the classes representing gelatinous forms, both cnidarians and ctenophores. The catchall groups Cnidarian and Ctenophora were not well resolved, while individual taxa within those groups (e.g. the ctenophore groups *Bolinopsis*, *Beroe*, and *Pleurobrachia*) were well classified. The catchall groups may have thus likely represented lower quality images (to both human and machine observers) that were more visually heterogeneous and possessed fewer useful features for identification.

Large, very deep CNNs benefit from large training sets (e.g. Cho *et al.*, 2016), and the training set used here is small compared with those used in contemporary machine vision research like ImageNet. It is however of similar size to several training sets used in plankton identification studies (order of hundreds to thousands of images per class: Hu and Davis, 2005; Bi *et al.*, 2015; Faillettaz *et al.*, 2016). The roughly lognormal size distribution of plankton populations makes finding less common taxa problematic. The size stratified technique used here attempted to balance the need to obtain examples of as many classes as possible, while not missing out on more rare forms. The classifier developed here discriminated several comparatively rare (and visually distinctive) taxa with high accuracy. An iterative process where the results of the classifier are checked and added to the training set will aid in producing a larger training set, but that does however leave open the possibility of an unknown bias being introduced to the network (i.e. the network probably classifies some images better than other and will bias towards those images). Examination of the unknown class and those removed by probability filtering will also be instructive, though in the case of the latter would involve looking through a very large image set (10^5 images in the case of the PWS image set so far) and would likely need to be subsampled. Training set size will continue to be problematic for plankton studies using imagery, every plankton imager has different optical characteristics, resolution, and lighting, which makes each image set different and not directly comparable. Presently, there are several large plankton training image

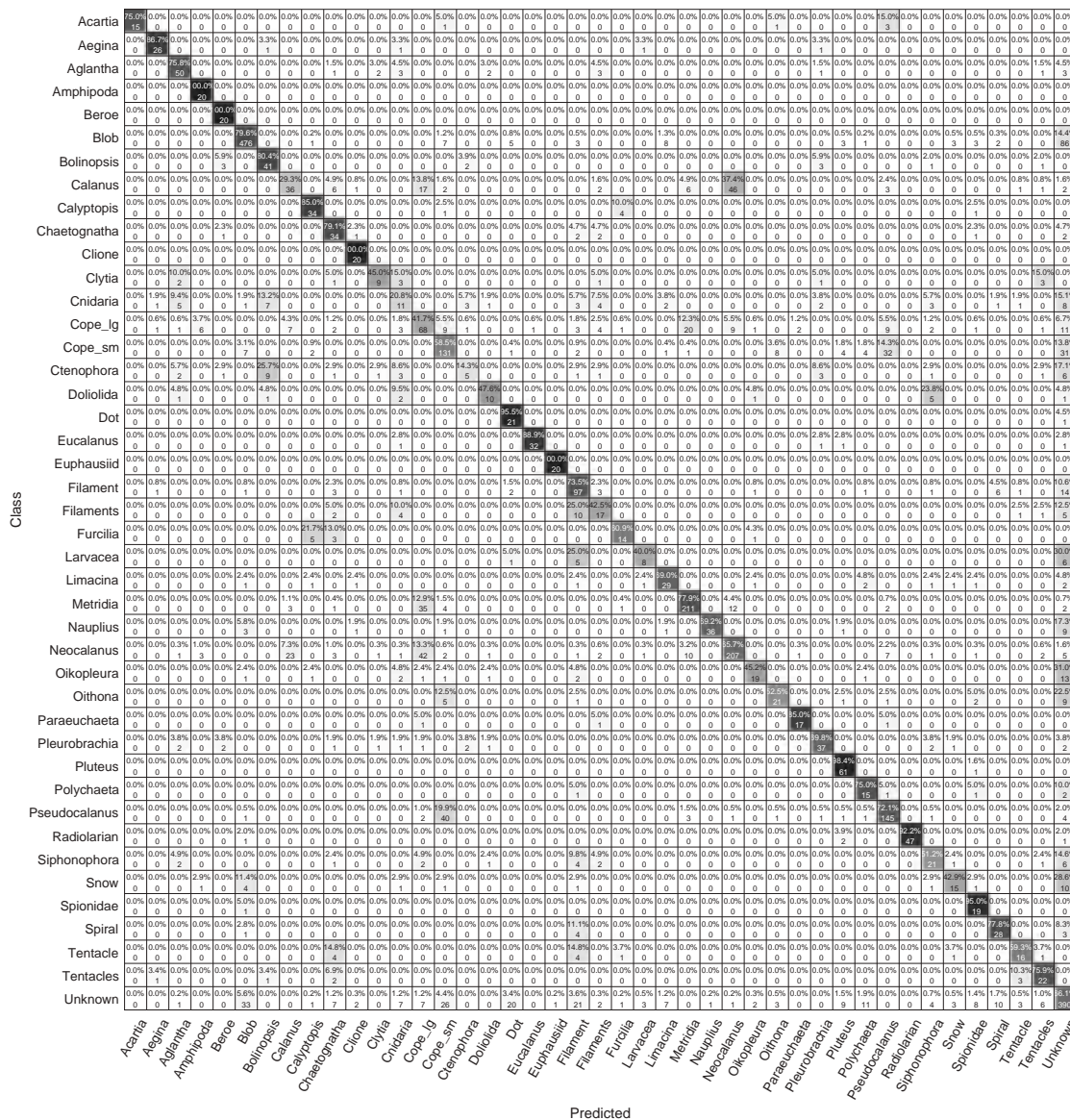


Figure 11. Confusion matrix for the classifier applied to the test set (i.e. images that the classifier did not experience while training).

sets available (Cowen et al., 2015; Orenstein et al., 2015) and, transfer learning, the use of networks pre-trained on other image sets has shown to improve the speed and accuracy of results (Orenstein and Beijbom, 2017; Rodrigues et al., 2018; ICES, 2020).

There is no panacea when approaching the problem of understanding zooplankton dynamics. Zooplankton are dilute, and a large volume of water must be sampled to obtain representative estimates of abundance. Plankton nets sample a large volume of water and allow fine scale taxonomic resolution but are expensive in terms of time and money and damage fragile taxa. Cameras sample a smaller volume of water and provide less taxonomic information but are inexpensive to operate following the initial capital outlay. Obtaining twice daily profiles over several months is simply not tractable with nets (Huntley and Lopez, 1992) but is with a camera. The classifier developed here permits high confidence the discrimination of several species-level and more broadly based groups.

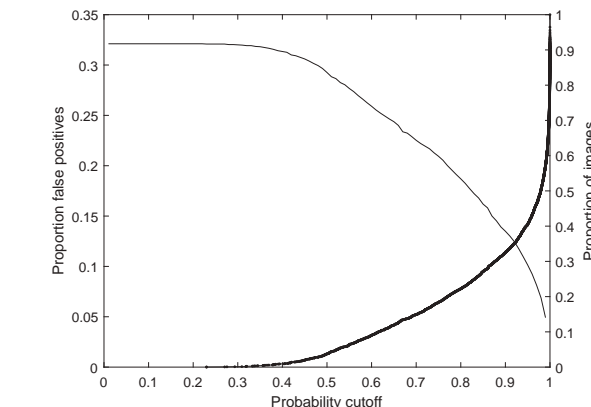


Figure 12. Proportion of false positives (black, left axis) and proportion of images that were rejected (red, right axis) as the probability cut-off was varied in the test set.

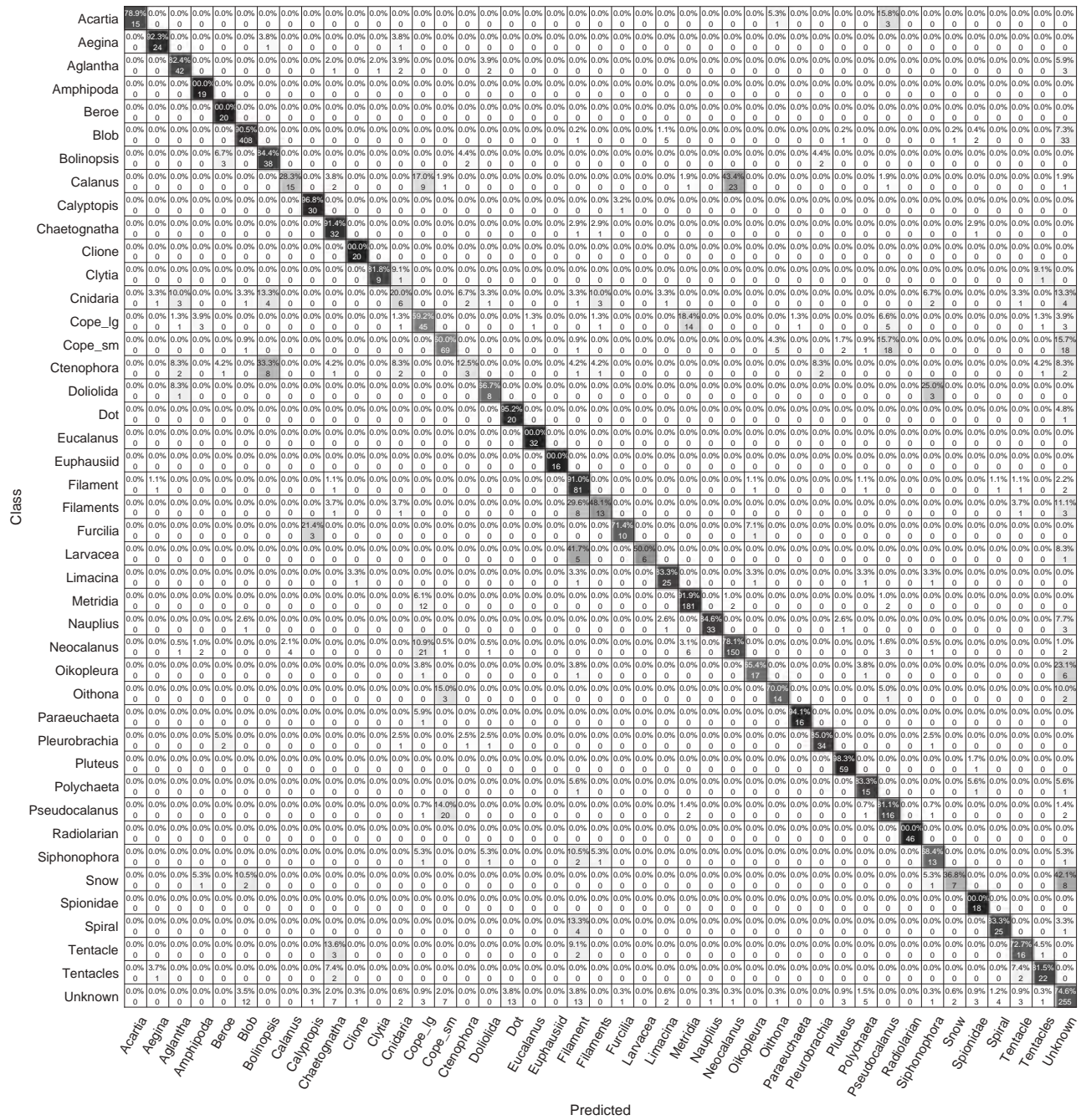


Figure 13. Confusion matrix for the classifier when applied to the test set and using a 90% probability threshold to discard uncertain classifications.

The value and usefulness of automatically classified imagery depends on the questions at hand. Simple information from zooplankton imagery such as abundance and size is easily determined with high confidence. For example, although there was some confusion between copepod species and the generalized copepod group, but if one is primarily interested in the abundance and relative biomass of copepods, that information may be determined with high confidence. Done over several years, estimates of zooplankton biomass could be of value to fisheries and ecosystem

managers (e.g. Möllmann et al., 2014). If there is interest in a single species, then more work may be required with the classified images to assure confidence but more inferential questions may be addressed.

Funding

This work was supported by the North Pacific Research Board [1502] and the Exxon Valdez Oil Spill Trustee Council [19120114-G]. The findings and conclusions presented by the

Table 4. Accuracy metrics for each class in the test set, when a 90% probability threshold was applied.

Class name	Precision	Recall	F1 score
Acartia	0.82	1	0.9
Aegina	0.92	1	0.96
Aglantha	0.93	0.86	0.9
Amphipoda	1	0.86	0.93
Beroe	1	0.83	0.91
Blob	0.91	0.92	0.92
Bolinopsis	0.91	0.78	0.84
Calanus	0.59	0.64	0.61
Calyptopis	0.93	0.97	0.95
Chaetognatha	0.92	0.9	0.91
Clione	1	1	1
Clytia	0.68	0.76	0.72
Cnidaria	0.26	0.41	0.32
Cope_lg	0.53	0.62	0.57
Cope_sm	0.58	0.63	0.6
Ctenophora	0.2	0.36	0.26
Doliolida	0.79	0.92	0.85
Dot	0.87	0.65	0.74
Eucalanus	1	0.97	0.98
Euphausiid	1	1	1
Filament	0.87	0.69	0.77
Filaments	0.71	0.73	0.72
Furcilia	0.92	0.86	0.89
Larvacea	0.86	0.75	0.8
Limacina	0.84	0.86	0.85
Metridia	0.95	0.87	0.91
Nauplius	0.96	0.92	0.94
Neocalanus	0.79	0.85	0.81
Oikopleura	0.88	0.75	0.81
Oithona	0.69	0.62	0.65
Paraeuchaeta	0.95	0.9	0.92
Pleurobrachia	0.83	0.81	0.82
Pluteus	0.98	0.95	0.97
Polychaeta	0.93	0.61	0.74
Pseudocalanus	0.8	0.78	0.79
Radiolarian	0.96	0.98	0.97
Siphonophora	0.84	0.87	0.85
Snow	0.68	0.83	0.75
Spionidae	0.88	0.88	0.88
Spiral	0.97	0.71	0.82
Tentacle	0.96	0.81	0.88
Tentacles	0.89	1	0.94
Unknown	0.77	0.79	0.78

Table 5. Comparison of the imaging specifications of published plankton imagers designed for zooplankton.

System	Imager resolution	Pixel resolution	Sampled volume	Illumination	References
CPICS	1 360 × 1 024	30 µm to 20 mm	1 ml	Darkfield	Grossmann <i>et al.</i> (2015)
VPR	Varies	30 µm to 5 cm	1.25–380 ml	Darkfield	Davis <i>et al.</i> (1992), Lombard <i>et al.</i> (2019)
ZOOVIS	2 448 × 2 050	10 µm	240 ml	Shadowgraph	Bi <i>et al.</i> (2013, 2015)
UVP	1 280 × 1 024	174 µm	1 020 ml	Light sheet	Picheral <i>et al.</i> (2010)
ISIIS	2 048 × 17 frames per second (line scan)	68 µm (in vertical)	169 l s ⁻¹	Shadowgraph	Cowen and Guigand (2008)
Zoocam	1 280 × 960	40 µm	250 ml	Shadowgraph	Ohman <i>et al.</i> (2019)
PWSPC	4 240 × 2 824	22.6 µm	450 ml	Darkfield	This project

authors are their own and do not necessarily reflect the views or position of the Trustee Council.

Acknowledgements

We thank Caitlin McKinstry for her help with the development of the training set; Bruce Rhoades for his considerable help with troubleshooting and integrating the camera with the profiler; Seth Adams for his assistance with programming the CNN; and the many technicians who assisted with deployment and service of the profiler. The Alaska Railroad Corporation generously donated train wheel anchors used in the deployment of the profiler. This manuscript was greatly improved by the comments of the anonymous referees.

References

- Abadi, A., Agarwal, P., Barham, E., Brevdo, Z., Chen, C., Citro, G. S., Corrado, A. *et al.* 2016. Tensorflow: large-scale machine learning on heterogeneous distributed systems. arXiv:1603.04467, 2016.
- Austen, G. E., Bindemann, M., Griffiths, R. A., and Roberts, D. L. 2017. Species identification by conservation practitioners using online images: accuracy and agreement between experts. PeerJ, doi: 10.7717/peerj.4157.
- Benfield, M., Grosjean, P., Culverhouse, P., Irigolen, X., Sieracki, M., Lopez-Urrutia, A., Dam, H. *et al.* 2007. RAPID: research on automated plankton identification. *Oceanography*, 20: 172–187.
- Benfield, M. C., Schwehm, C. J., Fredericks, R. G., Squires, G., Keenan, S. F., and Trevorrow, M. V. 2003. Measurements of zooplankton distributions with a high-resolution digital camera system. *Handbook of Scaling Methods in Aquatic Ecology: Measurement, Analysis, Simulation*, pp. 17–30. Ed. by L. Seuront and P. G. Strutton. CRC Press, Boca Raton.
- Bi, H., Cook, S., Yu, H., Benfield, M. C., and Houde, E. D. 2013. Deployment of an imaging system to investigate fine-scale spatial distribution of early life stages of the ctenophore *Mnemiopsis leidyi* in Chesapeake Bay. *Journal of Plankton Research*, 35: 270–280.
- Bi, H., Guo, Z., Benfield, M. C., Fan, C., Ford, M., Shahrestani, S., and Sieracki, J. M. 2015. A semi-automated image analysis procedure for in situ plankton imaging systems. *PLoS One*, 10: e0127121.
- Bridle, J. S. 1990. Probabilistic interpretation of feedforward classification network outputs, with relationships to statistical pattern recognition. *In Neurocomputing: Algorithms, Architectures and Applications* (1989). NATO ASI Series (Series F: Computer and Systems Sciences), 68, pp. 227–236. Ed. by F. F. Soulié and J. Héroult. Springer, Heidelberg.
- Bochinski, E., Bacha, G., Eiselein, V., Walles, T. J. W., Nejstgaard, J. C., and Sikora, T. 2019. Deep Active Learning for In Situ Plankton Classification. ICPR 2018 Workshops, LNCS 11188. doi: 10.1007/978-3-030-05792-3_1.
- Canny, J. 1986. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8: 679–698.
- Cheng, K., Cheng, X., Wang, Y., Bi, H., and Benfield, M. C. 2019. Enhanced convolutional neural network for plankton identification and enumeration. *PLoS One*, 14: e0219570.
- Cho, J., Lee, K., Shin, E., Choy, G., and Do, S. 2016. How much data is needed to train a medical image deep learning system to achieve necessary high accuracy? arXiv:1511.06348v2.
- Chollet, F. 2015. Keras. <https://keras.io/>.
- Cowen, R. K., and Guigand, C. M. 2008. In Situ Ichthyoplankton Imaging System (ISIIS): system design and preliminary results. *Limnology and Oceanography Methods*, 6: 126–132.
- Cowen, R. K., Sponaugle, S., Robinson, K., and Luo, J. (2015). PlanktonSet 1.0: plankton imagery data collected from F.G. Walton Smith in Straits of Florida from 2014-06-03 to 2014-06-06 and used in the 2015 National Data Science Bowl (NCEI Accession 0127422).
- Culverhouse, P. F., Simpson, R. G., Ellis, R., Lindley, J. A., Williams, R., Parisini, T., Reguera, B. *et al.* 1996. Automatic categorization of 23 species of dinoflagellate by artificial neural network. *Marine Ecology Progress Series*, 139: 281–287.
- Culverhouse, P. F., Williams, R., Reguera, B., Herry, V., and González-Gil, S. 2003. Do experts make mistakes? A comparison of human and machine identification of dinoflagellates. *Marine Ecology Progress Series*, 247: 17–25.
- Cui, J., Wei, B., Wang, C., Yu, Z., Zheng, H., Zhenk, B., and Yang, H. 2018. Texture and shape information fusion of convolutional neural network for plankton image classification. OCEANS-MTS/IEEE Kobe 10.1109/OCEANSKOBÉ.2018.8559156
- Davis, C. S., Gallager, S. M., Berman, M. S., Haury, L. R., and Strickler, J. R. 1992. The video plankton recorder (VPR): design and initial results. *Archiv für Hydrobiologie Beiheft Ergebnisse der Limnologie*, 36: 67–81.
- Faillietaz, R., Picheral, M., Luo, J. Y., Guigand, C., Cowen, R. K., and Irisson, J. O. 2016. Imperfect automatic image classification successfully describes plankton distribution patterns. *Methods in Oceanography*, 15–16: 60–77.
- Friedland, D. D., Stock, C., Drinkwater, K. F., Link, J. S., Leaf, R. T., Shank, B. V., Rose, J. M. *et al.* 2012. Pathways between primary production and fisheries yields of large marine ecosystems. *PLoS One*, 7: e28945.
- Goodfellow, I., Bengio, Y., and Courville, A. 2016. *Deep Learning*. MIT Press, Cambridge.
- Gorsky, G., Ohman, M. D., Picheral, M., Gasparini, S., Stemmann, L., Romagnan, J.-B., Cawood, A. *et al.* 2010. Digital zooplankton image analysis using the ZooScan integrated system. *Journal of Plankton Research*, 32: 285–303.
- Grossmann, M. M., Gallager, S. M., and Mitarai, S. 2015. Continuous monitoring of near-bottom mesoplankton communities in the East China Sea during a series of typhoons. *Journal of Oceanography*, 71: 115–124.
- Haralick, R. M. 1979. Statistical and structural approaches to texture. *Proceedings of IEEE*, 67: 786–804.
- Hassaballah, M., and Awad, A. I. (2016) Detection and description of image features: an introduction. *In Image Feature Detectors and Descriptors. Studies in Computational Intelligence*, 630, pp. 1–10. Ed. by A. Awad and M. Hassaballah Springer, Cham.
- Hendrycks, D., and Gimpel, K. 2018. A baseline for detecting misclassified and out-of-distribution examples in neural networks. arXiv: 1610.02136v3.
- Hu, Q., and Davis, C. 2005. Automatic plankton image recognition with co-occurrence matrices and support vector machine. *Marine Ecology Progress Series*, 295: 21–31.
- Huntley, M. E., and Lopez, M. D. 1992. Temperature-dependent production of marine copepods: a global synthesis. *The American Naturalist*, 140: 201–242.
- Jeffries, H. P., Berman, M. S., Poularikas, A. D., Katsinis, C., Melas, I., Sherman, K., and Bivins, L. 1984. Automated sizing, counting and identification of zooplankton by pattern recognition. *Marine Biology*, 78: 329–334.
- Johnson, K. S., Coletti, L. J., and Chavez, F. P. 2006. In situ ultraviolet spectrophotometry for high resolution and long-term monitoring of nitrate, bromide, and bisulfide in the ocean. *Deep Sea Research Part I*, 53: 561–573.
- King, M. 2007. *Fisheries Biology, Assessment and Management*, 2nd edn. Wiley-Blackwell, Oxford. 400 pp.
- Kingma, D. P., and Ba, J. 2015. Adam: a method for stochastic optimization. arXiv:1412.6980 [cs.LG].
- Krizhevsky, A., Sutskever, I., and Hinton, G. E. 2012. ImageNet classification with deep convolutional neural networks. *In Advances in neural information processing systems*, pp 1097–1105.

- Lee, H., Park, M., and Kim, J. 2016. Plankton classification on imbalanced large scale database via convolutional neural networks with transfer learning. *IEEE ICIP*. doi: 10.1109/ICIP.2016.7533053.
- Liu, J., Du, A., Wang, C., Yu, Z., Zheng, H., Zheng, B., and Zhang, H. 2018. Deep pyramidal residual networks for plankton image classification. *MTS/IEE OCEANS*. doi: 10.1109/OCEANSKOBE.2018.8559106
- Lombard, F., Boss, E., Waite, A. M., Vogt, M., Uitz, J., Stemann, L., Sosik, H. M. *et al.* 2019. Globally consistent quantitative observations of planktonic ecosystems. *Frontiers in Marine Science*. doi: 10.3389/fmars.2019.00196
- Longhurst, A. 2006. *Ecological Geography of the Sea*, 2nd edn. Academic Press, Cambridge. 560 pp.
- Luo, J. Y., Irisson, J.-O., Graham, B., Guigand, C., Sarafraz, A., Mader, C., and Cowen, R. K. 2018. Automated plankton image analysis using convolutional neural networks. *Limnology Oceanography Methods*, 16: 814–827.
- Mitra, A., Castellani, C., Gentleman, W. C., Jónasdóttir, S. H., Flynn, K. J., Bode, A., Halsband, C. *et al.* 2014. Bridging the gap between marine biogeochemical and fisheries sciences; configuring the zooplankton link. *Progress in Oceanography*, 129: 176–199.
- Möllmann, C., Lindegren, M., Blenckner, T., Bergström, L., Casini, M., Diekmann, R., Flinkman, J. *et al.* 2014. Implementing ecosystem-based fisheries management: from single-species to integrated ecosystem assessment and advice for Baltic Sea fish stocks. *ICES Journal of Marine Science*, 71: 1187–1197.
- Ohman, M. D., Davis, R. E., Sherman, J. T., Grindley, K. R., Whitmore, B. M., Nickels, C. F., and Ellen, J. S. 2019. Zooglider: an autonomous vehicle for optical and acoustic sensing of zooplankton. *Limnology Oceanography Methods*, 17: 686.
- Orenstein, E. C., Beijbom, O., Peacock, E. E., and Sosik, H. M. 2015. WHOI-plankton-a large scale fine grained visual recognition benchmark dataset for plankton classification. *CoRR* 2015; abs/1510.00745.
- Orenstein, E. C., and Beijbom, O. 2017. Transfer learning and deep feature extraction for planktonic image data sets. *In* 2017 IEEE Winter Conference on Applications of Computer Vision (WACV), Santa Rosa, CA, 2017, pp. 1082–1088. doi: 10.1109/WACV.2017.125.
- Pearson, K. 1904. *Mathematical Contributions to the Theory of Evolution on the Theory of Contingency and Its Relation to Association and Normal Correlation*. Dulau and Co., London. 34 pp.
- Perez, L., and Wang, J. 2017. The effectiveness of data augmentation in image classification using deep learning. arXiv:1712.04621.
- Picheral, M., Guidi, L., Stemann, L., Karl, D. M., Iddaoud, G., and Gorsky, G. 2010. *Limnology Oceanography Methods*, 8: 462–473.
- Rodrigues, F. C. M., Hirata, N. S. T., Abello, A. A., De La Cruz, L. T., Lopes, R. M., and Hirata, R. Jr 2018. Evaluation of Transfer Learning Scenarios in Plankton Image Classification. *VISIGRAPP*, doi: 10.5220/0006626703590366.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., *et al.* 2015. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115: 211–252.
- Samson, S., Hopkins, T., Remsen, A., Langebrake, L., Sutton, T., and Patten, J. 2001. A system for high resolution zooplankton imaging. *IEEE Journal of Oceanic Engineering*, 26: 671–676.
- Schröder, S.-M., Kiko, R., Irisson, J.-O., and Koch, R. 2018. Low-shot. *In* *Pattern Recognition*, pp. 391–404. 40th German Conference, 2018 Proceedings. Ed. by T. Brox, A. Bruhn. and M. Fritz. GCPR 2018 Stuttgart, Germany, October 9–12.
- Sheldon, R. W., and Parsons, T. R. 1967. A continuous size spectrum for particulate matter in the sea. *Journal of Fisheries Research Board of Canada*, 24: 909–915.
- Sosik, H. M., Peacock, E. E., and Brownlee, E. F. 2015. Annotated Plankton Images Data Set for Developing and Evaluating Classification Methods. <https://hdl.handle.net/10.1575/1912/7341>.
- Strickland, J. D. H., and Parsons, T. R. 1972. A practical handbook of seawater analysis. *Fisheries Research Board of Canada*, 167: 310.
- Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. 2015. Rethinking the inception architecture for computer vision, *CVPR*, vol. abs/1512.00567, arxiv.org/abs/1512.00567.
- Tajbakhsh, N., Shin, J. Y., Gurudu, S. R., Hurst, R. T., Kendall, C. B., Gotway, M. B., and Liang, J. 2016. Convolutional neural networks for medical image analysis: full training or fine tuning? *IEEE Transactions on Medical Imaging*, 35: 1299–1312.
- van Rijsbergen, C. J. 1979. *Information Retrieval*, 2nd edn. Butterworth, London.
- Wiebe, P. H., and Benfield, M. C. 2003. From the Hensen net toward four-dimensional biological oceanography. *Progress in Oceanography*, 56: 7–136.
- Yamashita, R., Nishio, M., Do, R. K. G., and Togashi, K. 2018. Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, 9: 611–629.

Handling editor: Cigdem Beyan