ICES Journal of Marine Science

ICES Journal of Marine Science (2018), doi:10.1093/icesjms/fsy147

Fish species identification using a convolutional neural network trained on synthetic data

Vaneeda Allken[1]*, Nils Olav Handegard[1], Shale Rosen[1], Tiffanie Schreyeck[2], Thomas Mahiout[2], and Ketil Malde[1,3]

[1]*Institute of Marine Research, P.O. Box 1870 Nordnes, N-5817 Bergen, Norway*
[2]*Department of Applied Mathematics and Modeling, Polytech Nice-Sophia, P.O. Box 145, 06903 Sophia Antipolis Cedex, France*
[3]*Department of Informatics, University of Bergen, P.O. Box 7803, N-5020 Bergen, Norway*

**Corresponding author: Tel: (+47) 55 23 85 00; e-mail: vaneeda@hi.no*

Allken, V., Handegard, N. O., Rosen, S., Schreyeck, T., Mahiout, T., and Malde, K. Fish species identification using a convolutional neural network trained on synthetic data. – ICES Journal of Marine Science, doi:10.1093/icesjms/fsy147.

Received 26 June 2018; revised 5 September 2018; accepted 6 September 2018.

Acoustic-trawl surveys are an important tool for marine stock management and environmental monitoring of marine life. Correctly assigning the acoustic signal to species or species groups is a challenge, and recently trawl camera systems have been developed to support interpretation of acoustic data. Examining images from known positions in the trawl track provides high resolution ground truth for the presence of species. Here, we develop and deploy a deep learning neural network to automate the classification of species present in images from the Deep Vision trawl camera system. To remedy the scarcity of training data, we developed a novel training regime based on realistic simulation of Deep Vision images. We achieved a classification accuracy of 94% for blue whiting, Atlantic herring, and Atlantic mackerel, showing that automatic species classification is a viable and efficient approach, and further that using synthetic data can effectively mitigate the all too common lack of training data.

Keywords: acoustic-trawl survey, deep learning, fish image classification, machine learning, trawl camera

Introduction

Sustainable exploitation of marine natural resources requires effective management based upon ongoing monitoring of the marine environment. Acoustic-trawl surveys (MacLennan and Simmonds, 2005) are one of the most important tools for assessing fish abundance. These are typically used for pelagic stocks, providing important input to the fisheries assessment models. When using calibrated echo sounders, fish density is related to backscattered energy (Foote, 1983) through the target strength (Foote, 1987). As target strength varies by species, correctly identifying the species detected acoustically is critical to correctly estimating fish density.

Acoustic-trawl surveys typically use trawl sampling to identify the species or species groups present. Trawl sampling only produces an aggregate collection of fish along the trawl path, and if different fish species are collected, assigning each species to specific locations can be challenging. Using camera equipment in the trawl is one way to increase the resolution along the trawl path. The Deep Vision (Scantrol Deep Vision AS, Bergen, Norway) system (Rosen and Holst, 2013) (Figure 1) funnels the trawl catch past a high-resolution stereoscopic camera chamber, before it is collected in the codend. Image pairs are taken with a fixed frequency of 5 or 10 frames per second, resulting in millions of images from a typical acoustic-trawl survey. Classification is challenging due to partial visible fish, fish at different orientations and shapes, and similarities between the species in terms of shape size and colouring. Each image is accompanied by information about GPS position, time, and depth.

Machine learning and computer vision techniques can be used to automate image processing, and tailored image recognition techniques have traditionally been developed to solve specific problems (LeCun *et al.*, 2015, and references therein). This is also the case for fish images, where specific techniques have been developed for species identification (White *et al.*, 2006) and fish segmentation (Chuang *et al.*, 2015), among others.

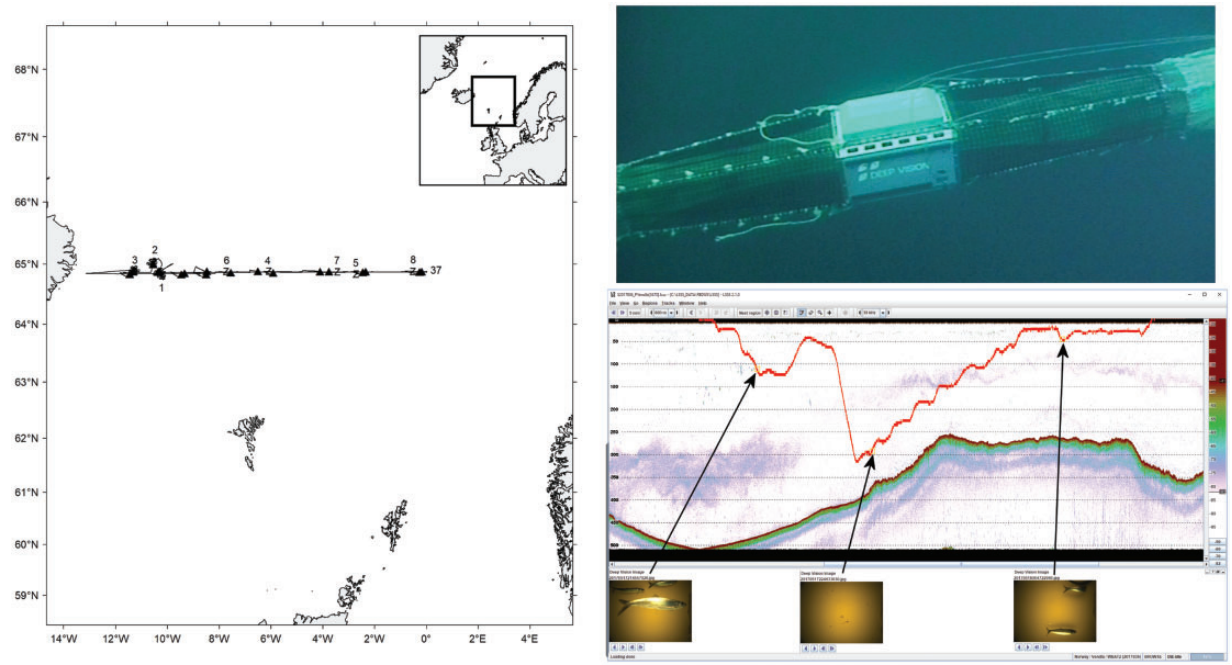

Figure 1. Transect at 65.83 degrees north latitude (left). Underwater image of Deep Vision system placed between trawl and codend (upper right). Trawl profile and images integrated with echosounder data in LSSS (lower right). The red line indicates the path of the trawl through the water column. Deep Vision images at bottom are from the positions indicated by the arrows and orange boxes along the trawl path.

Developing tailored solutions to specific problems is costly, but a promising technology has recently emerged in the form of *deep learning*, where deep convolutional neural networks learn to analyse the raw data directly (Schmidhuber, 2015). This technique has been particularly successful applied to image classification (Krizhevsky *et al.*, 2012; He *et al.*, 2016), but depends on having large sets of previously labelled data available.

Access to suitable training data is often limited, and several methods exist to mitigate this problem. Systems trained on a general or related problem can be adapted to the problem at hand by retraining parts (usually the final layer of a neural network), a process referred to as *transfer learning* (Sharif Razavian *et al.*, 2014; Yosinski *et al.*, 2014). Pre-trained networks have previously been used successfully to recognize fish species in images (Salman *et al.*, 2016; Siddiqui *et al.*, 2018).

Another commonly used option is to artificially expand the training set by applying transformations that preserve classes. This is referred to as *augmentation*. For instance, images can be randomly cropped, mirrored (Krizhevsky *et al.*, 2012), scaled, and rotated (Wan *et al.*, 2013). Other data types may require different transformations, the important property is that transformations do not affect the target variable. In recent years, more complex forms of data augmentation have emerged (Taylor and Nitschke, 2017), for example, using domain-specific synthesization or using *Generative Adversarial Networks* (GANs) (Goodfellow *et al.*, 2014). Here we implement a simulator that generates synthetic images to serve as training data. The images are composed randomly from components of real images, and the process is fast and straightforward.

Objectives

Our primary objective is to develop a system for automatic fish species identification to support acoustic-trawl surveys, using a state-of-the-art convolutional neural network for image classification. We target the long-running series of surveys of Norwegian spring spawning herring (*Clupea harengus* L.), the International Ecosystem Survey in the Nordic Seas (IESNS), which provides one of the main data series for the assessment of Norwegian Spring Spawning Herring. The survey covers a larger area of the Norwegian Sea, and is coordinated through the International Pelagic survey working group at the International Council for the Exploration of the Sea (ICES) with participation from Norway, Faroes, Iceland, Russia, and Denmark. Commonly encountered pelagic species in the area are herring, blue whiting (*Micromesistius poutassou*) and mackerel (*Scombrus scombrus*), and the difficulty of acoustically separating these species pose a central challenge.

A secondary objective is to explore the potential of the use of synthetic data as a way to generate the large datasets necessary for training deep learning classifiers. The lack of sufficient amounts of labelled training data limits the application of machine learning in many domains, and the process of labelling by a human expert is often labour intensive. Effective simulation methods to generate synthetic data thus opens up new fields to analysis by deep learning methods.

Material and methods

Data collection

In May 2017, a separate voyage was carried out back-to-back with the IESNS survey. The objective was to support the main survey by investigating potential sources of biases in the indices of abundance that are used in the assessment. The survey used the combined pelagic trawler and purse seiner FV Vendla, and followed the same transect as RV GO Sars from the Norwegian coast to the coast of Iceland, at 65.83 degrees north (Figure 1). Herring was found from 0 degrees longitude and west, with the highest

densities in the Icelandic zone, between 9 and 12 degrees west longitude. Mackerel was found east of 6 degrees west longitude, generally in the surface layers. Blue whiting was encountered at depth in trawling stations from the easternmost trawling position (1.4 degrees east longitude) to 8 degrees west longitude.

Trawl sampling was done using a Multpelt 832 pelagic trawl with 50 mm codend. In order to limit catches, a longitudinal split was put in the top of the codend extending 150–230 cm forward of the codline. The trawl was spread by 7 m Egersund SeaFlex trawl doors, with the upper hatches opened 50% and the lower hatches opened 12.5–25%. About 750 kg weights were fitted to the lower wing tips. Trawling speed was 3.5–5.0 knots, with lower speeds necessary when trawling at depth and highest speeds when trawling at the surface. The Deep Vision system was mounted between the trawl and codend (Figure 1). The Deep Vision images are read directly into the Large Scale Survey System (LSSS) software package used to discriminate acoustic data and the trawl's path through the water column is indicated on the echogram based upon depth and time stamp. LSSS corrects for the offset in distance between the vessel and trawl based upon the length of trawling warp and bridles.

During the survey, a total of 20 trawl stations were conducted using the Deep Vision system (Table 1). Thirteen of the hauls were long tows conducted with an open codend (average towing time 2 h 46 min), while seven were conducted with a closed codend but the split described above (average towing time with closed codend: 52 min). Images were collected for the entire time the Deep Vision system was in the water, a total of 1 216 914 stereo image pairs from 63 h 19 min of data collection. At one station, images were collected at 10 frames per second, but the system proved unstable at this frame rate (unable to maintain a constant frame rate and synchronization with the strobe, see below, was sometimes lost). The camera was stable at the frame rate of five images per second used at all other stations.

Table 1. Trawling data.

Longitude (deg)	Time of day (hh: mm UTC)	Depth (headrope, m)	Duration (hh: mm)	Image pairs (count)	Species
−11.6595	21: 13	0–300	03: 47	81 265	h
−11.2836	08: 41	0–260	00: 49	38 235	h
−10.5477	08: 52	0–200	00: 27	31 820	h
−10.4222	17: 58	0–160	00: 27	29 085	h
−10.2725	11: 30	0–280	03: 05	155 854[a]	h
−9.7074	08: 43	0–290	04: 04	88 645	h
−9.4059	14: 55	190–340	00: 35	27 065	h
−8.7934	10: 19	0–320	02: 09	56 410	h
−8.7926	19: 51	0–285	04: 13	90 970	h
−7.9564	15: 47	0–285	02: 39	62 980	h, bw
−6.7382	22: 12	0–200	01: 34	45 135	h, bw
−5.9108	09: 52	0–280	03: 48	87 550	h, bw, m
−4.6890	12: 15	0–350	03: 59	91 280	h, bw
−3.4793	06: 17	0–260	01: 36	42 630	h, bw, m
−2.6615	16: 58	0–290	03: 49	80 470	h, bw, m
−2.1604	01: 01	0–200	01: 03	32 705	h, bw, m
−0.8008	17: 35	0–370	03: 36	76 805	h, bw, m
−0.0419	23: 50	0-160	00: 50	27 315	h, bw, m
1.1279	19: 16	0–30	00: 38	29 240	m
1.4376	13: 43	200–360	00: 53	41 455	bw, m

Note: h, herring; m, mackerel; and bw, blue whiting.
[a]Deep Vision images were collected at 10 frames per second at one station, 5 frames per second at all other stations.

Dataset and image classification

From the images obtained from the survey, around a thousand images per species were manually curated such that only a single species (although often multiple individuals) were present. For acoustic surveys, we are interested in identifying the dominant species, and the pelagic fish we study here typically occur in monospecific schools. Splitting our dataset into a training, validation, and test set leaves us with a few hundred images per species, which may not be sufficient to train a deep neural network to optimal performance.

Generating synthetic images for training

In order to quickly provide a training set, we developed a system to generate N artificial images that are sufficiently similar to actual Deep Vision images to serve as training data. With the method described below, this system allows us to generate thousands of images using a small number of images of individual fish (Figure 2).

From the raw data, individual fish were selected and extracted manually using the Gimp lasso tool, resulting in a set of C cropped images of fish with a variety of orientations and sizes from each of the three species. Source images were selected where fish were fully visible (not occluded). In addition, we selected a set of 16 empty images (i.e. images with no fish or other objects) to serve as background. To generate an artificial image for a given species, the following procedure was used:

(1) A background image was randomly selected.
(2) One to six fish images were randomly selected from the cropped images for each species.
(3) Each cropped fish image was subjected to different transformations, e.g. random flipping, rotation and resizing (see Table 2).
(4) The fish images were pasted at random positions in size order, to emulate an approximate depth scaling.

Random positions were selected so that at least one-third of each fish was visible from the edges. To match the physical properties of the Deep Vision system, no more than one-third of a fish was allowed to extend below the bottom edge of the image, and fish were not allowed to extend above the top edge at all.

These parameters were chosen so that the images generated would bear the closest possible resemblance to the Deep Vision data. Generating one image took slightly less than 0.1 s. While the number of possible images is unlimited, generating and training on larger datasets takes more computational time. Since these images are generated from a fixed number of cropped images, one might expect diminishing returns beyond a certain number of generated images. Extracting cropped images of individual fish also requires a certain amount of manual effort. In order to explore how performance varies with the number of cropped fish images (C) and number of synthetic training images per species (N), we validated the network for different combinations of C and N.

The convolutional neural network classifier

We used the TensorFlow deep learning framework (Abadi *et al.*, 2015) with the Keras (Chollet and others, 2015) library. A Keras

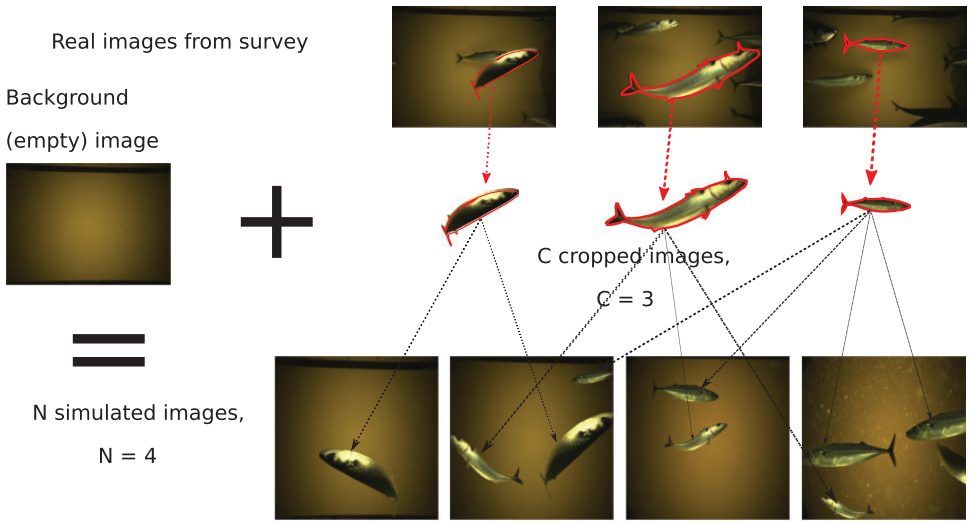

Figure 2. To generate images that realistically resemble Deep Vision photographs, instances of fish are cropped from real images and pasted onto empty background images at random positions, with random orientations and sizes. The number of fish per image varies between one and six. In this example, three individual mackerel images are extracted from Deep Vision images ($C = 3$) and used to generate four synthetic images ($N = 4$).

implementation of a state-of-the-art convolutional neural network model Inception 3 (Szegedy *et al.*, 2016) was used, with a classifier pretrained on the ImageNet classification dataset. To reduce the number of parameters and prevent overfitting, a global average pooling layer (Lin *et al.*, 2013) and a dropout layer (Srivastava *et al.*, 2014) were added. The last Inception layer is fully connected and consists of 1000 nodes, each representing a class in the ImageNet dataset. The outputs are normalized so that they sum to one using a softmax normalizer. This was replaced with a layer having three outputs, corresponding to the three fish species in our data. For training, we used cross-entropy loss as our objective, and minimized it using the RMSprop optimizer with a learning rate of 0.0001.

The whole network was finetuned on a new dataset consisting solely of images from one of the training sets described below, resized to (299, 299) for 10 epochs (complete iterations through the training data). The network was finetuned again on training images resized to (512, 512) for 50 epochs with early stopping, with a patience of 15 epochs, i.e. training was stopped either after the full 50 epochs or after 15 consecutive epochs (the patience) resulting in no improvement. The whole procedure including testing took approximately 45 min per dataset on average on an NVIDIA GeForce GTX 1080 Ti with Cuda 8.0 and cudnn 5.1.

Training datasets

We used 133 different training sets based on different configurations of C and N, where C varied between 10 and 70. For each C, 10 000 synthetic images per species were generated and we selected fractions of this dataset, such that the number of training images (N) varied between 100 and 1000 in steps of 100 and from 2000 to 10 000 in steps of 1000, resulting in 19*7 training sets.

In order to compare training on synthetic data with training on real images, we selected 10–70 images of each species from the available dataset and then trained the convolutional neural network with these images as training dataset. Following standard practice, we applied to the final images, both real and synthetic, a set of augmentation techniques that included rotation, translation, shearing, flipping, and zooming (see Table 3). To assess the importance of these transformations, we also trained the network on synthetic data without data augmentation.

Table 2. Table of transformations applied to the cropped images of individual fish used to generate the synthetic images.

Transformation method	Range
Prob. of left-right flip	1/3
Prob. of top-bottom flip	1/10
Size according to species	33/35 (h), 35/35 (m), 27/35 (bw)
Scale	Between 50/55 and 50/15
Rotation	Gaussian, mean = 0, $SD = 8$

Note: h, herring; m, mackerel; and bw, blue whiting.

Validation and testing

For validation and testing, we used a balanced dataset (with the same number of images for each fish species) consisting of a total of 3000 images obtained from the Deep Vision survey. To avoid overfitting on the training data, 20% of this dataset was kept as validation data to monitor the training process after each epoch, with images previously unseen by the network. The trained network for a given epoch was saved whenever validation loss decreased. After training, we tested our network on a test dataset consisting of 2400 real images, 800 from each species, using the saved model corresponding to the minimum validation loss. For each image, the predicted fish species is the class with the maximum softmax output.

Results

We conducted a series of experiments where different number of training images (N) were generated from varying numbers of individual cropped out fish images (C). The training images were used to train the convolutional neural network. Each network thus trained was tested on the test dataset of 2400 images, and the resulting classification accuracy was recorded (Figure 3). Test

accuracy is expressed here as the percentage of predictions that match the ground truth label.

The best accuracy (94.1%) on the test dataset was achieved when training on a dataset consisting of 15 000 images (5000 per species), based on 70 cropped images. Because some weights are randomly initialized in the network, running an experiment several times with otherwise same parameters can give slightly different results. To test how choices of C and N affect accuracy, we fitted a multiple linear regression model to logit transformed observed accuracies ($R^2 = 0.57$, $F_{(2,\ 130)} = 87$, $p < 0.1$). It was found that the number of individual fish images (C) significantly predicted the accuracies ($\beta = 0.74$, $p < 0.01$). The number of images (N) also came out as a significant, albeit smaller, effect ($\beta = 0.16$, $p < 0.01$). It is worth noting that when we ran the same regression on the accuracies from networks that were trained *without* data augmentation on the synthetic images, the effect of N on the accuracy doubled (see Supplementary Figure S1).

In order to provide a baseline for comparing the efficacy of training on synthetic data, we also trained the classifier on the same number of *real* images, which were subjected to the same data augmentation as the synthetic data (Table 4). The test accuracy for classifiers trained on real images varied from 50.8% to 71.1%.

Where the classifier fails

With the most accurate classifier ($C = 70$ and $N = 5000$), 141 out of 2400 images (5.9% of the test set) are misclassified (Table 5). Of the three species, the most commonly misclassified species is the mackerel in this instance. Note that this does not necessarily constitute a trend across classifiers. The next best classifier, for instance, overpredicts blue whiting and misclassifies herring most frequently.

To better understand the potential sources of error, the misclassified images were manually inspected and categorized as shown in Figure 4. More than 50% of the inaccurately identified fish images contained only parts of the fish. The rest of the misclassified images contained for the most part, either multiple fish or fish orientated such that features important for identification were obscured.

The classifier outputs its predictions in the form of a softmax output, assigning a number to each class and scaling the output so that the final prediction scores sum to one. To determine the classification, the class with the highest prediction score is chosen, but the prediction scores can also be interpreted as the classifier's confidence in the classification. For three categories, the maximum prediction score is between 0.33 (representing approximately equal likelihood of the three categories) and 1.0 (representing complete confidence in the classification). The distribution of prediction scores is shown in Figure 5 (left), along with the fraction of misclassified images for each prediction score category. As expected, the fraction of misclassified images decreases with increasing confidence threshold. By setting a minimum threshold we can substantially improve the accuracy of the remaining predictions, but at the cost of leaving some images unclassified (Figure 5, right).

Discussion

We have shown that a standard convolutional neural network is able to correctly identify fish species with up to 94% accuracy on a dataset of images collected from a standard fisheries survey using a commercially available camera system. Misclassified images are mainly caused by fish seen only partially or in non-ideal orientations. Nevertheless, the system successfully identifies a large number of images where fish are only partially seen, or where fish are partially occluded. These situations often confound methods that rely on segmentation of fish by identifying the outline (White *et al.*, 2006; Chuang *et al.*, 2015).

Table 3. Table of augmentation techniques.

Augmentation method	Range
Rotation range	40
Width shift range	0.2
Height shift range	0.2
Rescale	1/255
Zoom range	0.2
Horizontal flip	True

Table 4. Test accuracy obtained on the 2400 test images when the network was trained on real images vs. test accuracies for the best combination of N for synthetic images.

Number of images used	10	20	30	40	50	60	70
Real images	50.8	53.0	67.1	64.4	63.3	71.1	67.2
Synthetic data (best)	89.2	89.4	91.5	92.2	92.4	93.6	94.1

Notes: Each column indicates the number of real labelled images or cropped single images used. The first row shows the test accuracy after using 10–70 real images for training, and the second row shows the best test results across N when using 10–70 individual fish crops to generate the training images.

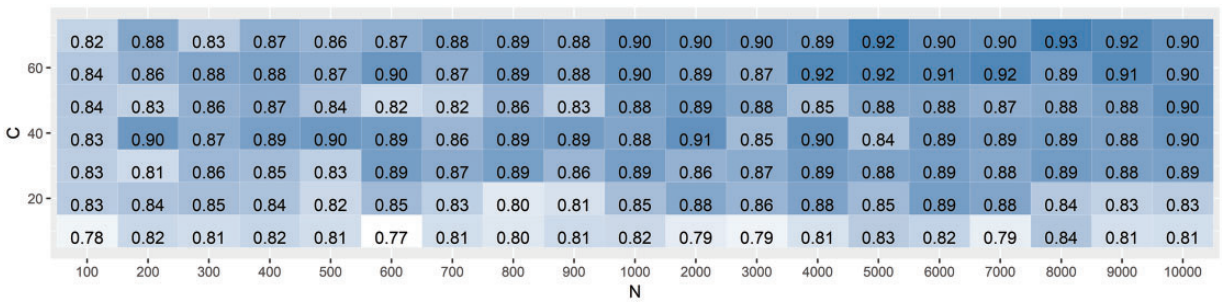

Figure 3. Test accuracy as a function of C and N after training on synthetic images.

The strength of neural networks is that general architectures and programming libraries are readily available, and that they are flexible enough to be applied to new problems with only small modifications. Here we have used a pre-trained neural network as a starting point and retrained it on our own data in a process called transfer learning. The pre-trained network is able to identify low-level features useful for image classification, and we were able to quickly and effectively adapt it from its original purpose of differentiating between the 1000 object classes in the ImageNet dataset to identifying fish species.

Importantly, by using synthetic images we were able to achieve a very high accuracy with a surprisingly small number of labelled images. This allows us to easily generate thousands of realistic images with higher variation than with traditional augmentation methods. The uniformity of the Deep Vision images, with their regular background and lighting, makes them a good fit for this approach. Training with synthetic data have previously been used successfully in other contexts. For instance, Jaderberg *et al.* (2016) generated synthetic images to train a system to retrieve textual information. Similarly, Ødegaard *et al.* (2016) used

Table 5. Confusion matrix obtained using best classifier, with C = 70 and N = 5000 with data augmentation.

Predicted	Blue whiting	Herring	Mackerel	Total, predicted
Actual				
Blue whiting	771	16	11	798
Herring	29	765	66	860
Mackerel	0	19	723	742
Total, actual	800	800	800	2400

Note: The columns show the predictions for each species, e.g. out of 800 blue whiting, 771 are correctly classified whereas 29 are misclassified as herring.

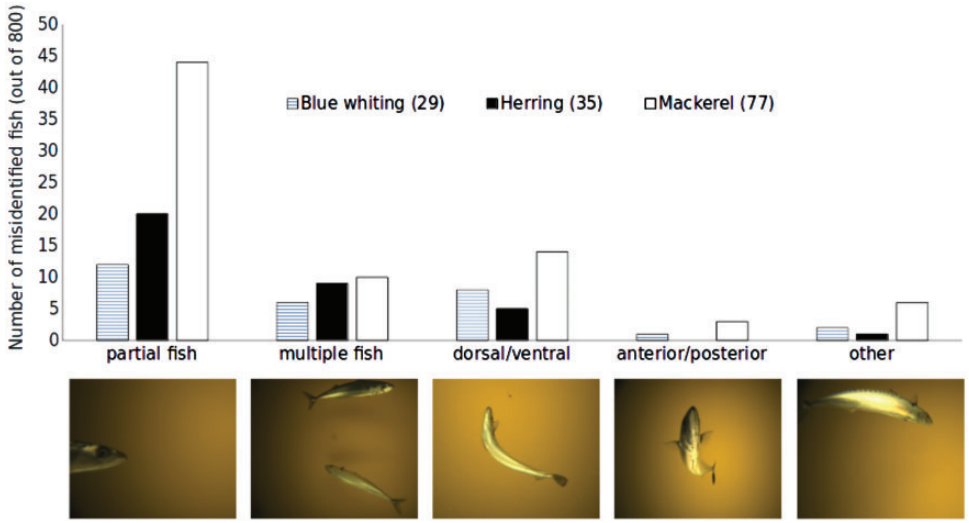

Figure 4. Characteristics of images where the convolutional neural network technique assigned incorrect species. Values in parentheses indicate total number of images in which fish were misidentified (out of 800 images per species). Images below each category are examples where species was misidentified.

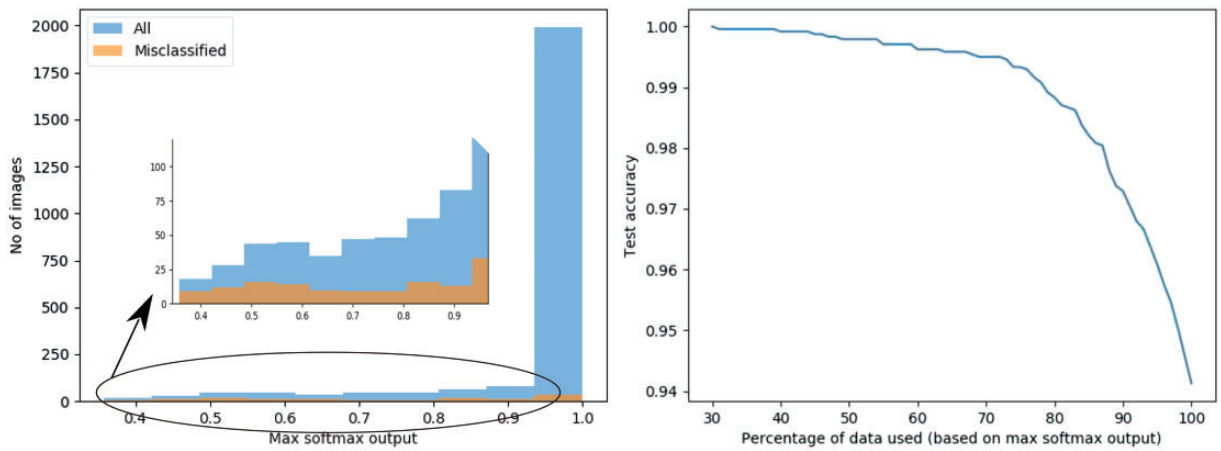

Figure 5. Maximum softmax output as an indicator of accuracy. *Left:* Distribution of test images as a function of prediction score (maximum softmax output). The fraction of images misclassified for different softmax outputs are shown in orange. *Right:* Test accuracy as function of the percentage of data retained, based on their prediction scores (images with the lowest prediction scores are discarded). Accuracy increases when the images with the lowest prediction scores are removed from the dataset.

synthetic radar signatures to train a convolutional network to recognize ships. For many important problems there is a lack of adequate amounts of labelled data for training, and generating synthetic data is likely to be a central technique to mitigate this. It is important to ensure that the images generated are tailored to the problem at hand and to avoid introducing bias in the selection process.

Acoustic-trawl surveys provide indices of abundance to stock assessment models, and correctly allocating acoustic energy to species is a key challenge. Typically, trawl catches are used to aid the process but lacks spatial resolution. Adding a camera to the trawl provides necessary resolution along the trawl path (Rosen and Holst, 2013), and, in cases where echogram regions (c.f. Figure 1) are classified to single species class, predicting the major class along the trawl track is sufficient. This is the rationale for classifying the images rather than individual fish within the images. This also allows us to filter out images with low confidence classification, and interpolating between high quality images is likely to result in improved accuracy and more reliable region classification.

Trawl-camera systems allow trawling with an open codend, reducing the impact on the environment as well as on fish welfare. This is relevant in cases where physical samples are not needed, and when the current practice is too selective or destructive such that, e.g. small objects, are not retained in the catch.

Several extensions of the work are planned. The three species used here are central to the herring assessments, but future development will extend this to differentiate more species. For applications where we are interested in the mix of species within an image, typically for demersal trawl surveys, the one-species per image approach may not be sufficient. Alternatively, the network could be trained to predict mixed categories, or extended to classify each individual fish. The latter would require segmentation of individual fish and could be achieved by deep learning segmentation techniques (Girshick *et al.*, 2014; Long *et al.*, 2015; Redmon *et al.*, 2016; Badrinarayanan *et al.*, 2017). Segmentation also allows us to efficiently use the stereoscopic information in the Deep Vision system, and next steps include using stereoscopic information to determine fish size and condition.

We are now proceeding to integrate the Deep Vision into our processing pipelines, using the IESSNS survey as a use case. This includes software that can be deployed operationally to provide predictions, adjustments to the software that is used to classify the acoustics (LSSS), and hardware adjustments to ensure proper handling of data. It is only after these steps have been taken that the full value of the development will be realized.

Supplementary data

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

Acknowledgements

This project was funded in part by Research Council of Norway projects 270966/O70 (COGMAR) and 203477 (CRISP). The data were collected through the REDUS project with funding from the Norwegian Ministry of Trade, Industry and Fisheries. We are grateful for the collaboration with Scantrol Deep Vision, and we thank the crew on FV Vendla for skilful handling of the Deep Vision system during the voyage. We also thank the anonymous reviewers for their helpful suggestions.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., and Chen, Z. 2015. TensorFlow: large-scale machine learning on heterogeneous systems. Software available from https://www.tensorflow.org/.

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.

Chollet, F. and others. 2015. Keras 2.1.3. https://keras.io.

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. Ed. by J. Zhou, X. Bai and T. Caelli. IGI Global. IGI Global, Hershey, PA. 79–106 pp.

Foote, K. G. 1983. Linearity of fisheries acoustics, with addition theorems. The Journal of the Acoustical Society of America, 73: 1932–1940.

Foote, K. G. 1987. Fish target strengths for use in echo integrator surveys. Journal of the Acoustical Society of America, 82: 981–987.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. *In* Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Colombus, OH, USA, pp. 580–587.

Goodfellow, I. J., Shlens, J., and Szegedy, C. 2014. Explaining and harnessing adversarial examples, pp. 1–11. ISSN: 0012-7183. arXiv: 1412.6572. http://arxiv.org/abs/1412.6572.

He, K., Zhang, X., Ren, S., and Sun, J. 2016. Deep residual learning for image recognition. *In* 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, pp. 770–778.

Jaderberg, M., Simonyan, K., Vedaldi, A., and Zisserman, A. 2016. Reading text in the wild with convolutional neural networks. International Journal of Computer Vision, 116: 1–20.

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. http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf.

LeCun, Y., Bengio, Y., and Hinton, G. 2015. Deep learning. Nature, 521(7553): 436–444.

Lin, M., Chen, Q., and Yan, S. 2013. Network in network. arXiv preprint: 101312.4400. http://arxiv.org/abs/1312.4400.

Long, J., Shelhamer, E., and Darrell, T. 2015. Fully convolutional networks for semantic segmentation. *In* Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, pp. 3431–3440.

MacLennan, D., and Simmonds, E. 2005. Fisheries Acoustics. Fish and Aquatic Resources Series 10. Chapman & Hall, London.

Ødegaard, N., Knapskog, A. O., Cochin, C., and Louvigne, J. C. 2016. Classification of ships using real and simulated data in a convolutional neural network. *In* 2016 IEEE Radar Conference (RadarConf), Philadelphia, PA, USA, pp. 1–6.

Redmon, J., Divvala, S., Girshick, R., and Farhadi, A. 2016. You only look once: unified, real-time object detection. *In* Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Killarney, Ireland, pp. 779–788.

Rosen, S., and Holst, J. C. 2013. DeepVision in-trawl imaging: sampling the water column in four dimensions. Fisheries Research, 148: 64–73.

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.

Schmidhuber, J. 2015. Deep learning in neural networks: an overview. Neural Networks, 61: 85–117.

Sharif Razavian, A., Azizpour, H., Sullivan, J., and Carlsson, S. 2014. Cnn features off-the-shelf: an astounding baseline for recognition. *In* Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops, Columbus, OH, USA, pp. 806–813.

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.

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.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. 2016. Rethinking the inception architecture for computer vision. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, Nevada, pp. 2818–2826.

Taylor, L., and Nitschke, G. 2017. Improving deep learning using generic data augmentation. arXiv: 1708.06020.http://arxiv.org/abs/1708.06020.

Wan, L., Zeiler, M., Zhang, S., Cun, Y. L., and Fergus, R. 2013. Regularization of neural networks using DropConnect. *In PMLR*, Atlanta, Georgia, USA, pp. 1058–1066. http://proceedings.mlr.press/v28/wan13.html.

White, D. J., Svellingen, C., and Strachan, N. J. C. 2006. Automated measurement of species and length of fish by computer vision. Fisheries Research, 80: 203–210.

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. http://papers.nips.cc/paper/5347-how-transferable-are-features-in-deep-neural-networks.pdf.

Handling editor: Richard O'Driscoll