Deep neural networks for analysis of fisheries surveillance video and automated monitoring of fish discards. G. French, M. Mackiewicz, M. Fisher, H. Holah, R. Kilburn, N. Campbell, and C. Needle, ICES Journal of Marine Science (2019), **doi:10.1093/icesjms/fsz149**

Deep Neural Networks for Analysis of Fisheries Surveillance Video and Automated Monitoring of Fish Discards

Geoff French, Michal Mackiewicz, Mark Fisher
School of Computing Sciences, University of East Anglia, Norwich, UK
g.french, m.mackiewicz, m.fisher@uea.ac.uk

Helen Holah, Rachel Kilburn, Neil Campbell, Coby Needle
Marine Laboratory, 375 Victoria Road, Aberdeen, Scotland
Helen.Holah, Rachel.Kilburn, Neil.Campbell, Coby.Needle@gov.scot

July 10, 2019

Abstract

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: deep learning, computer vision and CCTV

1 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 in order to ensure compliance with the trawlers individual quota. Quota is assigned according to the total allowable catch (TAC) 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, measuing 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; firstly to detect and count fish leaving the discard chute and secondly 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).

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 (Russakovsky et al., 2015, Lin et al., 2014).

We review a body of prior work in Section 2. 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 detail in Section 3.

We use instance segmentation to isolate individual fish within an image. This component is discussed in Section 4. 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 Section 5.

In order 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 Section 6.

The future directions of this work can be found in Section 7.

2 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 Section 1.

2.1 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. (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 approximately 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 (e.g. (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 initialise the network layers (Sun et al., 2016, Qin 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.

2.2 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 tranfer 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 maximise 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 maximising performance on the real-life data. In surveillance situations where data is 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.

2.3 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 invididal 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 (e.g. 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) in order 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 resoltuion 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.

3 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 in order 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.

3.1 Video sources

The surveillance footage was captured in 800p HD resolution and stored in MPEG-4 format. The videos come from 5 sources; 4 commercial fishing vessels and 1 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 personel 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 Section 5.2. The footage is summarised in Table 1. Example frames are shown in Figure 1.

Vessel	Type	# 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			

Table 1: Summary of video footage

3.2 Web application

To facilitate collaboration between Marine Scotland and University of East Anglia personnel, a web application was developed using the Django Framework[1]. The website allows Marine Scotland staff to upload CCTV footage and annotate images for training our computer vision systems (see Section 3.4.2). It was extended to support the inter-observer species identification variability experiment discussed in Section 5.3.

[1]Available from `https://djangoproject.com`

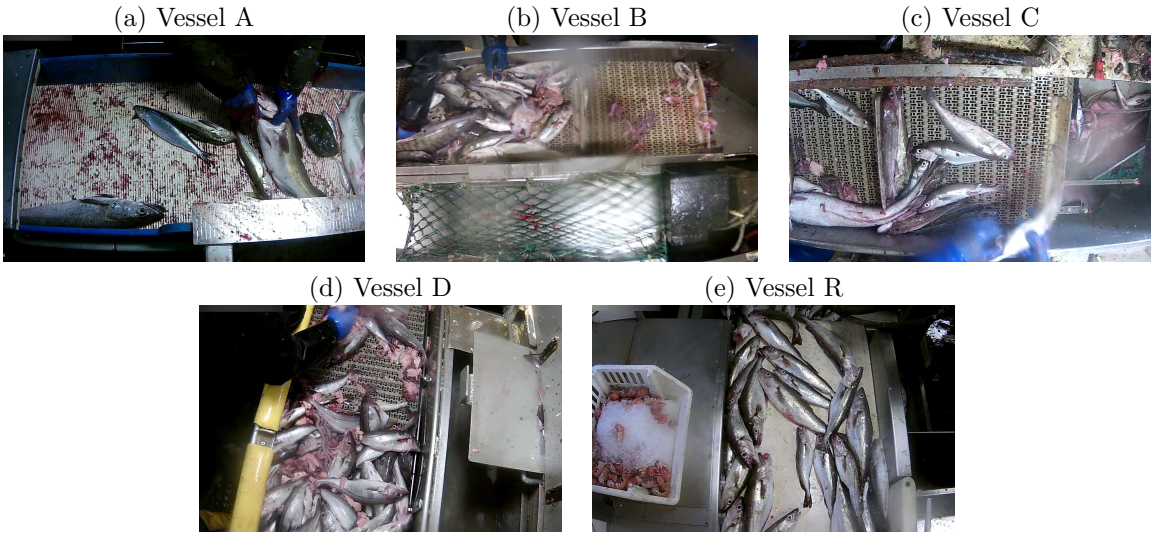

Figure 1: Images from each vessel

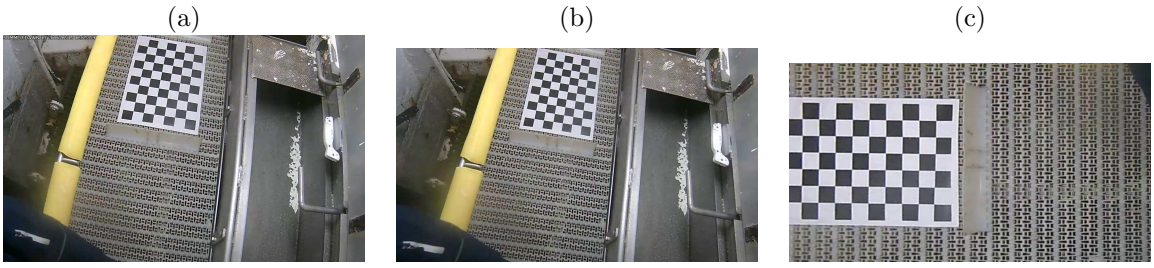

Figure 2: The belt extraction and calibration process: (a) checkerboard on belt, (b) with lens distortion removed and (c) with perspective warp used to transform belt into rectilinear space and exterior cropped out

3.3 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.

3.3.1 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 co-ordinates of the checkerboard corners, generating a map that specifies which cells are covered (`histogram2D` in the algorithm). If more than 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
```

3.3.2 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 mis-aligment and crop the region corresponding to the belt.

3.3.3 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 co-ordinates 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 less than 2 milliseconds on a desktop machine.

3.4 Segmentation and species ID training set

A segmentation data set consisting of still frames extracted from the video footage was required in order 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.

3.4.1 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 (ECC) 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 computing correlation using RGB or greyscale 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 $\frac{1}{16}$ resolution. Correlation between `pool3` at $\frac{1}{8}$ resolution were 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 (Chintala et al.).

3.4.2 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[2] allows the user to draw polygonal annotations and classify them. 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 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

[2]Available from `http://bitbucket.org/ueacomputervision/image-labelling-tool`

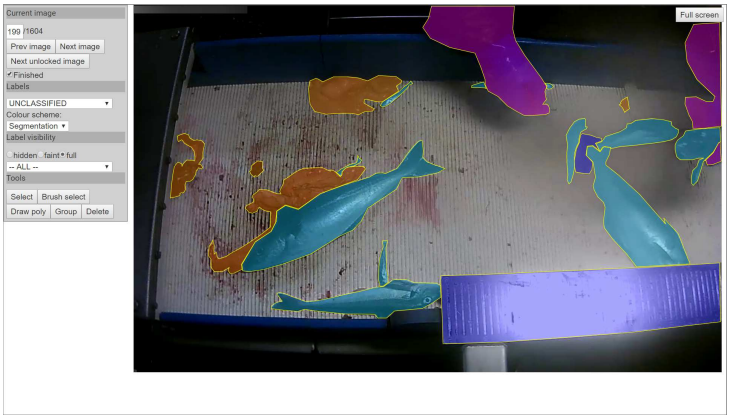

Figure 3: Web-based segmentation annotation tool

a new and more accurate segmentation model, resulting in a cyclic process.

Vessel	# annotated images	# annotated fish
Vessel A	204	1459
Vessel B	263	1254
Vessel C	145	1588
Vessel D	153	4809
Vessel R	137	1498
Total	902	10608

Table 2: Segmentation training set

3.4.3 Data

The training data for the segmentation system consists of 902 annotated frames drawn from videos from the 5 vessels and is summarised in Table 2. While many more frames were extracted, this is the subset that has been annotated so far.

4 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 in order 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[3]. As stated in Section 2 it

[3]We use the COCO (Lin et al., 2014) pre-trained implementation of Mask R-CNN provided by the

Figure 4: Instance segmentation applied to a frame from footage from Vessel R (research vessel); the blue labels are generated automatically

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

As stated in Section 3.4.2 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 maximised the quality of the automatically generated annotations using test-time augmentation (He et al., 2017); each image was segmented 8 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.

4.1 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 in order to select the class-specific bounding box regressor and mask head to optimise 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 Section 5.1 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.

torchvision library that is developed by the PyTorch (Chintala et al.) team. It produces good results and trains quickly.

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.

4.2 Training procedure

Data augmentation artifically 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 segmentaiton 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 to 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 initialised later layers and 10^{-5} for the pre-trained layers that come from the *torchvision* (Chintala et al.) Mask R-CNN implementation. We randomly crop 512×512 pixel regions from our rectilinear belt images and build mini-batches of crops from 4 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 more than 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.

5 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.

5.1 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 Section 4).

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 7 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 disgracting cues from parts of the image outside the bounds of the fish.

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 3: Summary of species identification dataset from commercial vessels

	Cod	Haddock	Whiting	Saithe	Hake	Monk	Mackerel
# of fish	1451	12482	14068	861	304		1837
# of videos	3	18	13	2	2		1

	Horse mackerel	Norway pout	Plaice	Long rough dab	Common dab	Grey gurnard	Red gurnard
# of fish	496	5574	2402	1495	1601	1599	65
# of videos	1	2	3	1	2	3	1

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

5.2 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 summarised in Table 4. This further illustrates the advantage of seprating segmentation and species classification into separate steps, as mentioned in Section 4.1.

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 were 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 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.

5.3 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 4-fold cross validation. As stated in Section 3.4 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 the proportion of samples mis-predicted as belonging to other classes. Perfect performance is indicated by 100% along the diagonal and 0% everywhere else.

5.3.1 Train and test on research samples

The research footage covers 13 species out of the 14 considered in this project. We don't 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.

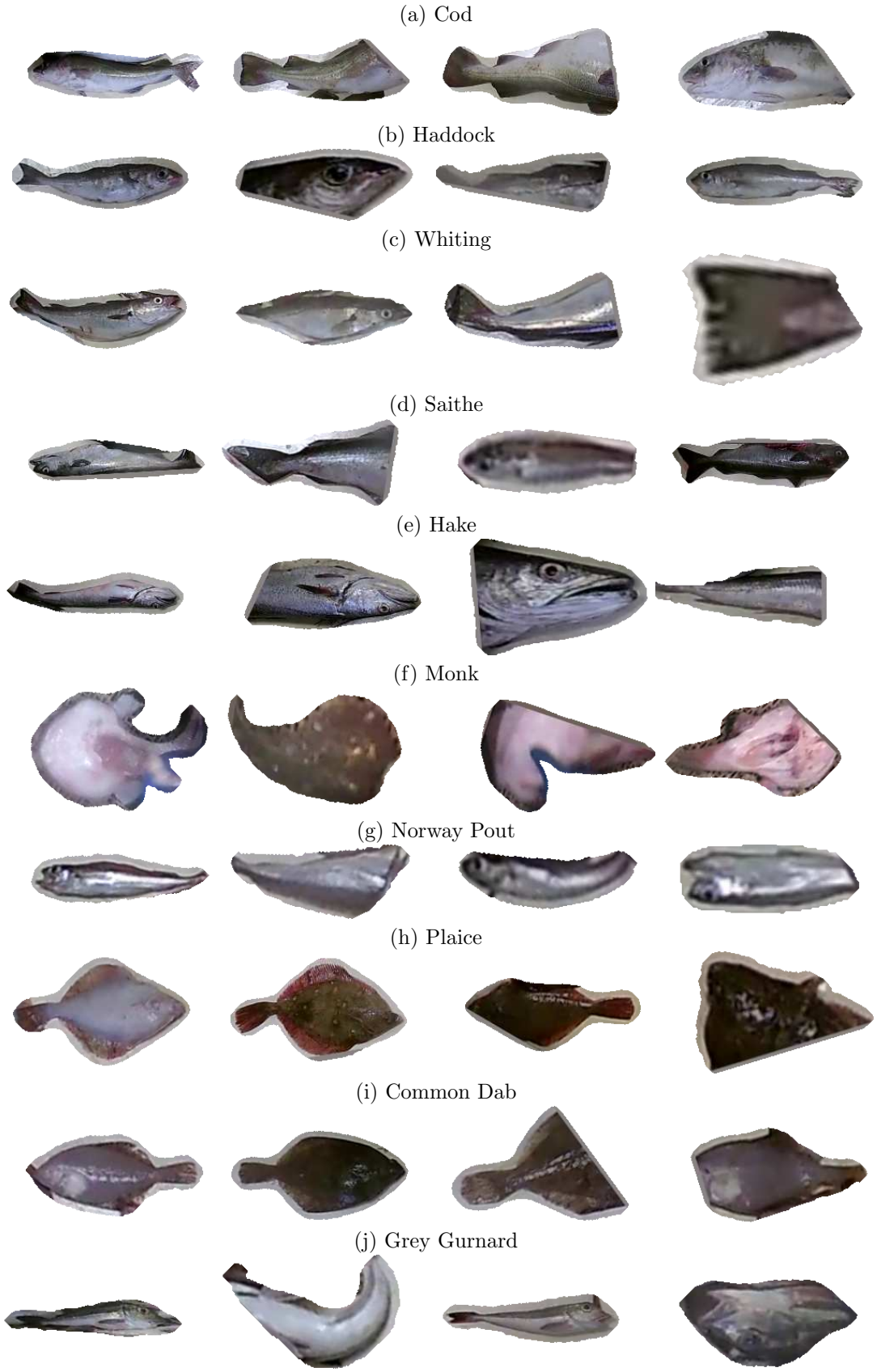

Figure 5: Examples from the species identification dataset. All fish were from the single species research vessel footage, apart from monk which were taken from commercial footage. Samples were chosen to illustrate that the classifier often receives only a partial fish or one whose orientation hides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

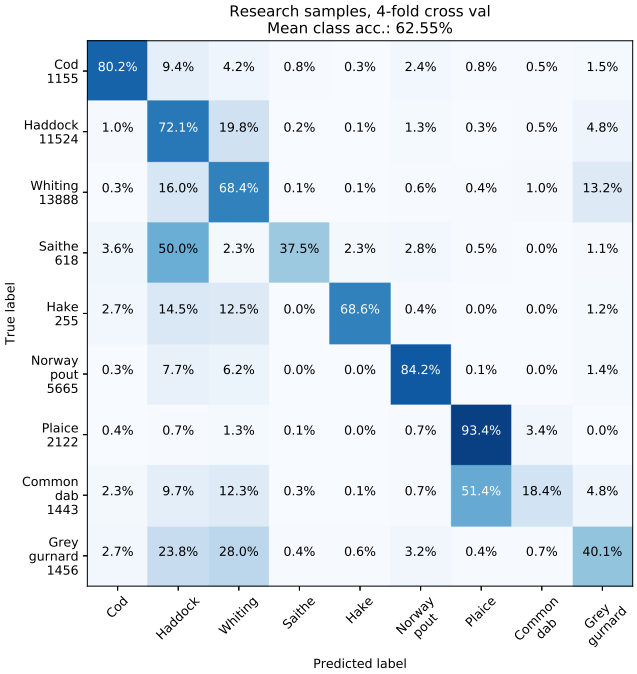

Figure 6: Confusion matrix for research samples, 4-fold cross validation

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 prioritise difficult un-annotated samples for manual annotation (Wang and Shang, 2014).

5.3.2 Train and test on commercial samples

Figure 7 (a) 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 approximately 20 times as many as there are commercial – incurrs 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 scarity of whiting in the commercial samples.

5.3.3 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

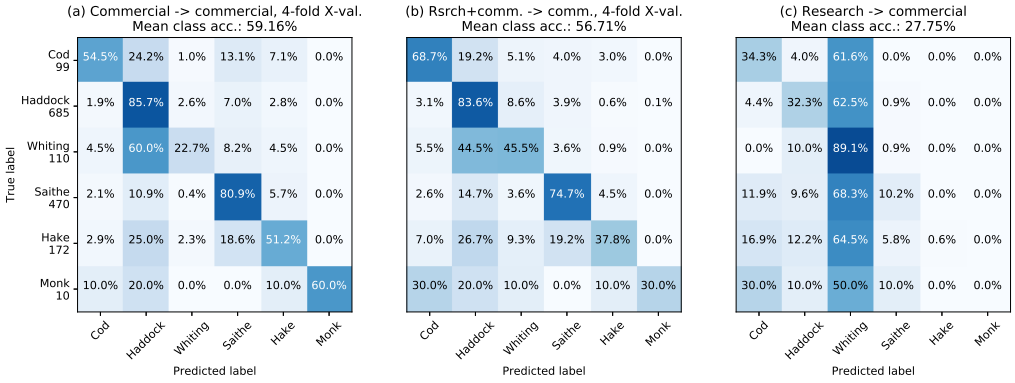

Figure 7: Confusion matrices for (a) train and test on commercial (4-fold cross validation), (b) train on research and commercial, test on commercial (4-fold cross validation) and (c) train on research, test on commercial. Without the monk class the mean class accuracies are (a) 59%, (b) 62.05% and (c) 33.3%

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.

5.3.4 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 and hake and testing on the commercial samples is shown in Figure 7 (c). Comparing the performance between (a) and (c) illustrates the effect of the domain gap; in spite of the fact that there are approximately 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.

6 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.

250 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 in order 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 in order to ensure sufficient representation for the purpose of meaningful analysis. Given the class imbalance present in our data (see Table 3) we used the ex-

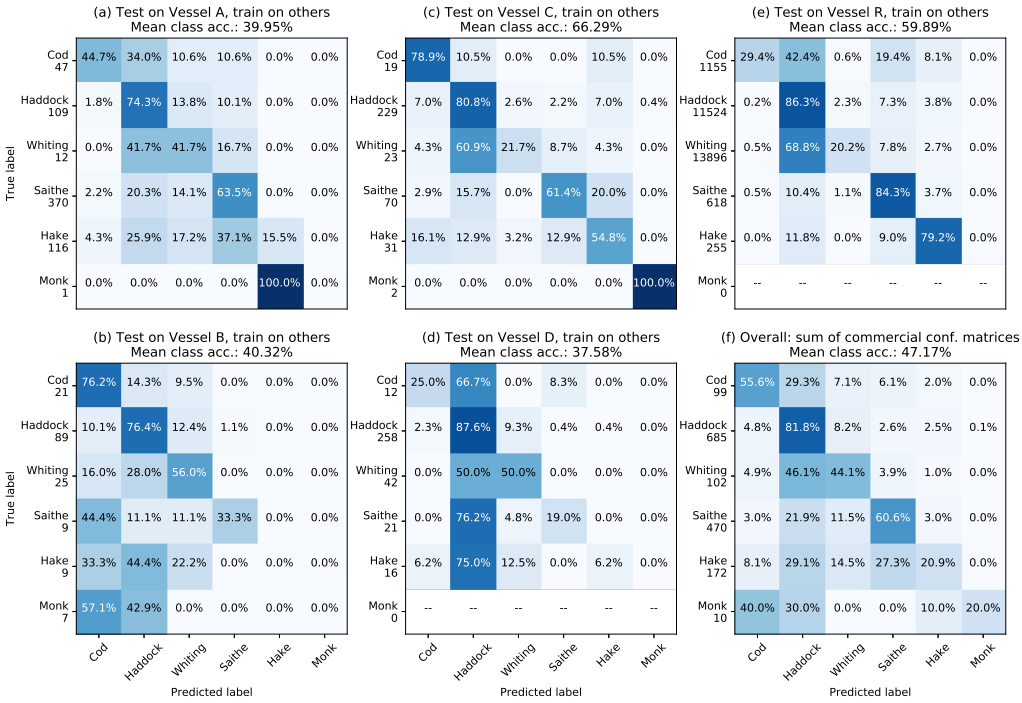

Figure 8: Performance when evaluating on samples from one vessel while training on others. Overall performance the result of computing the sum of the other confusion matrices. Overall mean class accuracy without under-represented monk class is 52.6%.

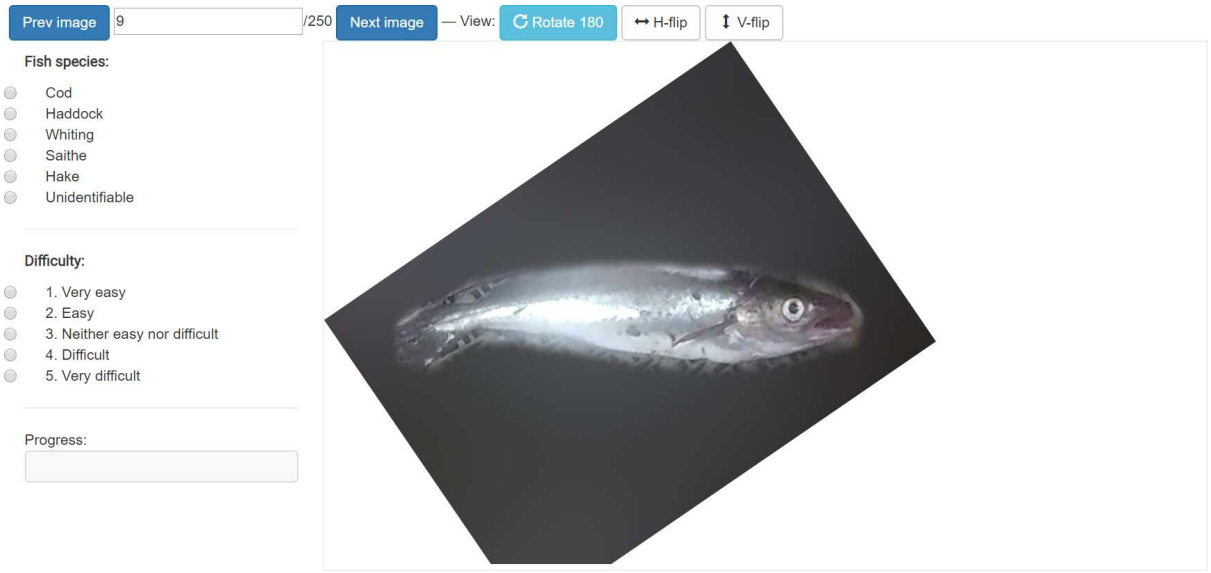

Figure 9: Inter-observer variability species identification tool as seen by the participants

isting species annotations to select samples for the dataset. While these individual fish had been previously annotated by Marine Scotland staff who later partitipated 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. 50 samples were selected from the remaining 5 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.

6.1 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.

6.2 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 Section 5.3.3, 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 out-performing observer 6, the lowest scoring human observer.

7 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 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

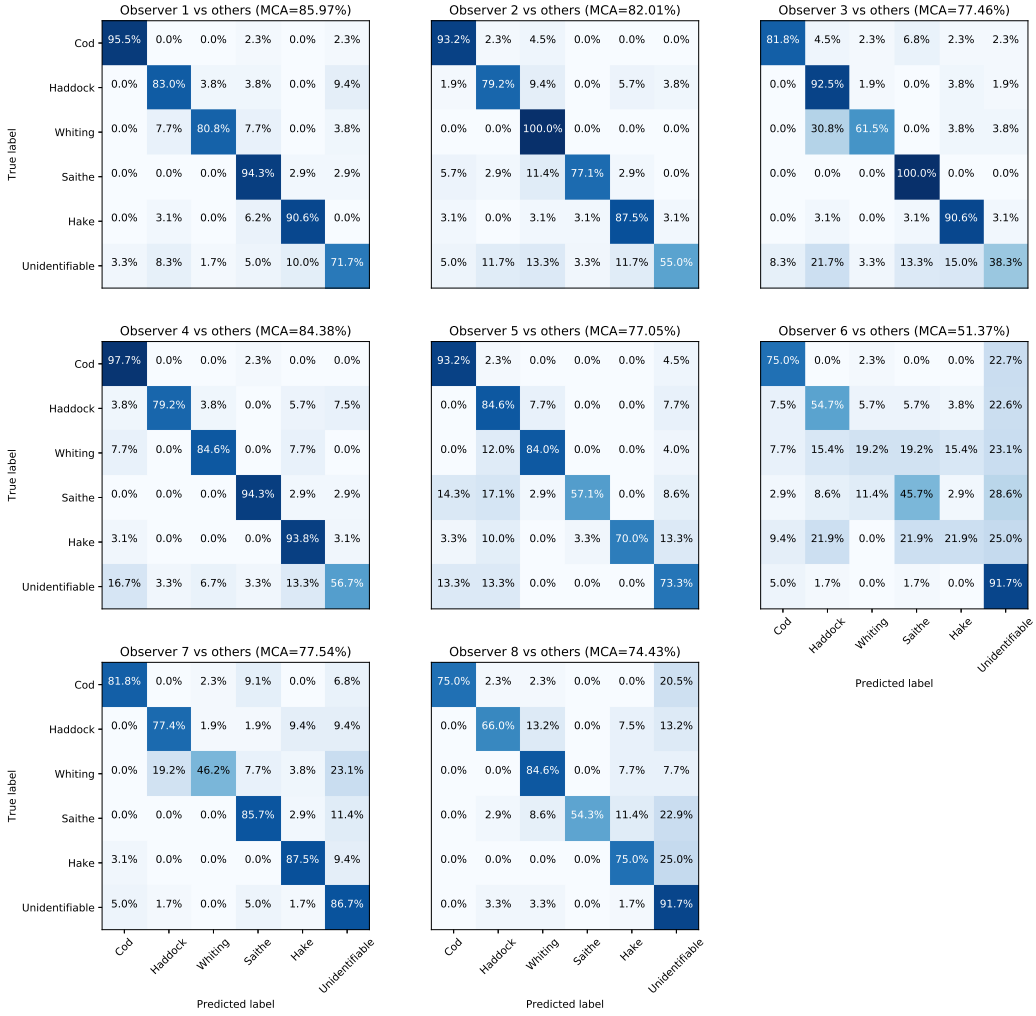

Figure 10: Inter-observer agreement confusion matrices. Each confusion matrix compares the species choice of an observer with the majority vote of the other seven observers.

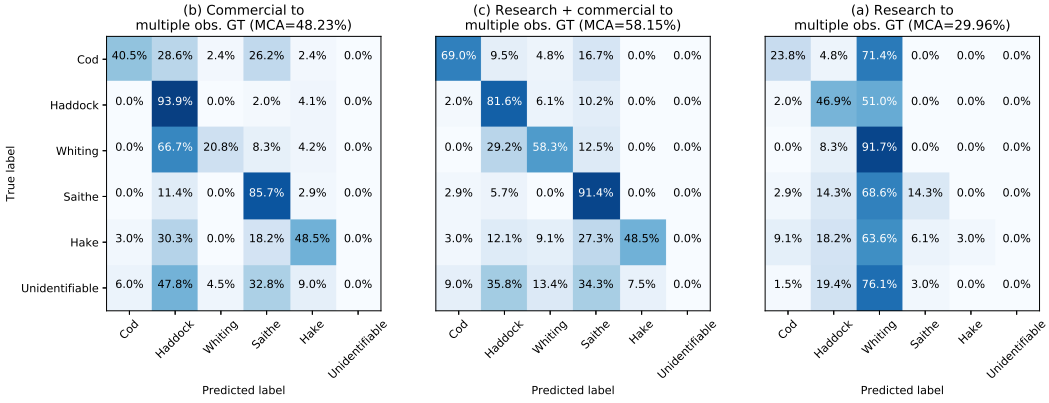

Figure 11: Classifier predictions in comparison to those of the expert observers

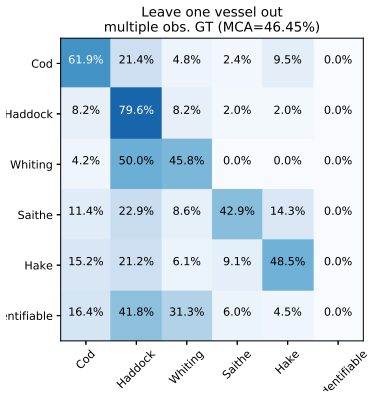

Figure 12: Classifier predictions in comparison to those of the expert observers; evaluate on samples from one vessel while training on samples from others

temporarily due to occlusion. There is a signficant body of work on the topic of person re-identification (e.g. (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 Section 5.3.1 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 7 (a) and (b)). We believe that growing the number of annotated 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 un-annoated samples and using it to prioritise them for manual annotation, optimising 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 this domain gap can be seen by comparing the results presented in Figure 7 (c) and Figure 7 (a). 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 un-annoated samples and using it to prioritise them for manual annotation, optimising 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.

This work was funded under the European Union Horizon 2020 SMARTFISH project, grant agreement no. 773521.

We would like thank nVidia coportation for their generous donation of a Titan X GPU.

References

M. K. S. Alsmadi, K. B. Omar, S. A. Noah, and I. Almarashdah. 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(2):215–221, 2009. URL `http://arxiv.org/abs/0912.0986`.

S. Beucher and F. Meyer. The morphological approach to segmentation: the watershed transformation. Mathematical morphology in image processing. *Optical Engineering*, 34:433–481, 1993.

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

G. Bradski. OpenCV. *Dr. Dobb's Journal of Software Tools*, 2000.

S. Chintala et al. Pytorch. URL `http://pytorch.org`.

M.-C. Chuang, J.-N. Hwang, and K. Williams. A feature learning and object recognition framework for underwater fish images. *IEEE Transactions on Image Processing*, 25(4): 1862–1872, 2016.

J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655, 2014.

G. D. Evangelidis and E. Z. Psarakis. Parametric image alignment using enhanced correlation coefficient maximization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(10):1858–1865, 2008.

G. French, M. Fisher, M. Mackiewicz, and C. Needle. 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*, 2015.

G. French, M. Mackiewicz, and M. Fisher. Self-ensembling for visual domain adaptation. In *International Conference on Learning Representations*, 2018. URL `https://openreview.net/forum?id=rkpoTaxA-`.

P. Guo and R. Farrell. 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)*, pages 1876–1885. IEEE, 2019.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2980–2988. IEEE, 2017.

B. G. Hu, R. Gosine, L. X. Cao, and C. de Silva. Application of a fuzzy classification technique in computer grading of fish products. *Fuzzy Systems, IEEE Transactions on*, 6(1):144–152, Feb 1998. ISSN 1063-6706. doi: 10.1109/91.660814.

J. Hu, D. Li, Q. Duan, Y. Han, G. Chen, and X. Si. Fish species classification by color, texture and multi-class support vector machine using computer vision. *Comput. Electron. Agric.*, 88:133–140, Oct. 2012. ISSN 0168-1699. doi: 10.1016/j.compag.2012.07.008.

T. Kluyver, B. Ragan-Kelley, F. Pérez, B. E. Granger, M. Bussonnier, J. Frederic, K. Kelley, J. B. Hamrick, J. Grout, S. Corlay, et al. Jupyter notebooks-a publishing format for reproducible computational workflows. In *20th International Conference on Electronic Publishing*, pages 87–90, 2016.

A. Krizhevsky, I. Sutskever, and G. E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems 25*, pages 1097–1105. 2012.

M. Li, X. Zhu, and S. Gong. Unsupervised person re-identification by deep learning tracklet association. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 737–753, 2018.

T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

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

J. R. Mathiassen, E. Misimi, M. Bondø, E. Veliyulin, and S. O. Østvik. Trends in application of imaging technologies to inspection of fish and fish products. *Trends in Food Science & Technology*, 22(6):257 – 275, 2011. ISSN 0924-2244. doi: 10.1016/j.tifs.2011.03.006.

C. L. Needle, R. Dinsdale, T. B. Buch, R. M. D. Catarino, J. Drewery, and N. Butler. Scottish science applications of remote electronic monitoring. *ICES Journal of Marine Science: Journal du Conseil*, 2014.

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

S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems 28*, pages 91–99. 2015.

O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3): 211–252, 2015. doi: 10.1007/s11263-015-0816-y.

K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *European conference on computer vision*, pages 213–226. Springer, 2010.

K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

F. Storbeck and B. Daan. Fish species recognition using computer vision and a neural network. *Fisheries Research*, 51(1):11 – 15, 2001. ISSN 0165-7836.

N. J. C. Strachan. Recognition of fish species by colour and shape. *Image and Vision Computing*, pages 2–10, 1993.

X. Sun, J. Shi, J. Dong, and X. Wang. Fish recognition from low-resolution underwater images. In *2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, pages 471–476. IEEE, 2016.

I. Tayama, M. Shimdate, N. Kubuta, and Y. Nomura. Application of optical sensor for fish sorting. *Refigeration*, 57(661):1146–1150, 1982.

S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, T. Yu, and the scikit-image contributors. scikit-image: image processing in Python. *PeerJ*, 2:e453, 6 2014. ISSN 2167-8359. doi: 10.7717/peerj.453. URL `https://doi.org/10.7717/peerj.453`.

D. Wang and Y. Shang. A new active labeling method for deep learning. In *2014 International joint conference on neural networks (IJCNN)*, pages 112–119. IEEE, 2014.

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

S. Xie and Z. Tu. Holistically-nested edge detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1395–1403, 2015.

Z. Yang, T. Luo, D. Wang, Z. Hu, J. Gao, and L. Wang. Learning to navigate for fine-grained classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 420–435, 2018.

Z. Zheng, C. Guo, X. Zheng, Z. Yu, W. Wang, H. Zheng, M. Fu, and B. Zheng. Fish recognition from a vessel camera using deep convolutional neural network and data augmentation. In *2018 OCEANS-MTS/IEEE Kobe Techno-Oceans (OTO)*, pages 1–5. IEEE, 2018.

B. Zion. The use of computer vision technologies in aquaculture - A review. *Computers and Electronics in Agriculture*, 88:125–132, 2012.