AutoFish: Dataset and Benchmark for Fine-grained Analysis of Fish

Bengtson, Stefan Hein; Lehotský, Daniel; Ismiroglou, Vasiliki; Madsen, Niels; Moeslund, Thomas B.; Pedersen, Malte

Published in:
Proceedings - 2025 IEEE Winter Conference on Applications of Computer Vision Workshops, WACVW 2025

Publication date:
2025

Document Version
Accepted author manuscript, peer reviewed version

AutoFish: Dataset and Benchmark for Fine-grained Analysis of Fish

Stefan Hein Bengtson[1,2], Daniel Lehotský[1], Vasiliki Ismiroglou[1,2], Niels Madsen[3]
Thomas B. Moeslund[1,2], and Malte Pedersen[1,2]

[1]Visual Analysis and Perception Lab, Aalborg University, Denmark
[2]Pioneer Centre for AI, Copenhagen, Denmark
[3]Section of Biology and Environmental Science, Aalborg University, Denmark

Abstract

Automated fish documentation processes are in the near future expected to play an essential role in sustainable fisheries management and for addressing challenges of overfishing. In this paper, we present a novel and publicly available dataset named AutoFish designed for fine-grained fish analysis. The dataset comprises 1,500 images of 454 specimens of visually similar fish placed in various constellations on a white conveyor belt and annotated with instance segmentation masks, IDs, and length measurements. The data was collected in a controlled environment using an RGB camera. The annotation procedure involved manual point annotations, initial segmentation masks proposed by the Segment Anything Model (SAM), and subsequent manual correction of the masks. We establish baseline instance segmentation results using two variations of the Mask2Former architecture, with the best performing model reaching an mAP of 89.15%. Additionally, we present two baseline length estimation methods, the best performing being a custom MobileNetV2-based regression model reaching an MAE of 0.62cm in images with no occlusion and 1.38cm in images with occlusion. Link to project page: https://vap.aau.dk/autofish/.

1. Introduction

Earth's marine ecosystems are facing an unprecedented threat, in large part due to overfishing. Among its most significant consequences are habitat destruction, loss of biodiversity, and ecological imbalances in marine environments, profoundly impacting coastal communities worldwide. Consequently, there has been a surge in research focused on scalable solutions for monitoring marine environments and turning the tide [22, 34].

The traditional approach to fisheries management, relying on catch limits and periodic onshore inspections, has proven insufficient in curbing overfishing and ensuring sustainable fisheries and healthy marine environments [12]. As a result, there is a pressing need to adopt innovative and technology-driven solutions that can effectively address the challenges posed by overfishing. Real-time automated monitoring of catch compositions could enhance compliance, reduce manual reporting for fishermen, and provide authorities reliable data for sustainable fisheries management [2]. A streamlined data collection and decision making could foster a transparent and responsible fishing industry, promoting ecological preservation while supporting the socio-economic well-being of coastal communities [25].

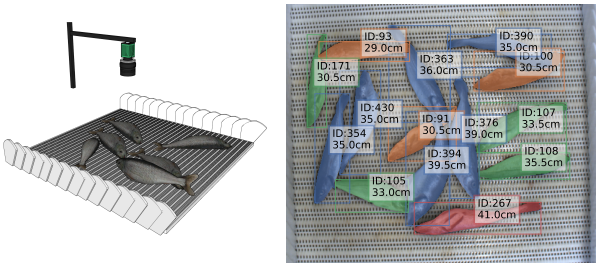

Figure 1. Illustration of the recording setup and an example image from the *AutoFish* dataset with an overlay of groundtruth bounding boxes, instance segmentations, IDs, and lengths.

In this work, we address the lack of image data from the fishing industry by proposing a novel dataset and baseline results related to automated documentation of catch compositions. Our contributions are the following:

- *AutoFish*, a publicly available dataset for fine-grained analysis of fish, with instance segmentation masks, IDs and manual length measurements for every specimen.
- A novel group-based data acquisition method for avoiding cross-contamination between data splits.
- Baseline results for instance segmentation and length estimation methods on the *AutoFish* dataset.

2. Related work

Research in automated catch monitoring systems based on computer vision has been conducted for years and, traditionally, species identification and length estimation models have been based on handcrafted features [1, 31, 32, 37, 39]. However, their design makes them poor at handling unforeseen objects, and it is not trivial to train the models to new species, as they may require an entire new set of handcrafted features or additional parameter tuning. Recently, there has been a paradigm shift in catch monitoring from handcrafted features to learned features [2], facilitated by the introduction of effective deep convolutional neural networks (CNNs) for image classification [9, 16]. However, image classification models do not directly take the spatial location of the object into account and is generally not suited for handling scenes with multiple fish. Nonetheless, image classification has been used for tasks, like identifying the presence of tuna or billfish from electronic monitoring (EM) cameras monitoring the deck of fishing vessels [20].

When handling multiple fish, it is beneficial to know the position (object detection) and species (object classification) of every fish present in the image. The YOLO [26] architecture has been a popular choice for detecting and classifying fish using bounding box representations. A YOLOv3 model was used by van Essen et al. [35] to locate and classify fish on conveyor belts and Sokolova et al. [30] used a modified YOLOv5 with an additional regression output to detect, classify, and estimate the weight of fish on conveyor belts. However, bounding boxes are not able to accurately capture the shape of non-rigid objects like fish that can bend and deform, or depict occlusion details in cases of overlap. In these cases, it is typically preferred to represent the fish on pixel-level using segmentation masks. French et al. [5] were among the first to propose methods for monitoring fish on conveyor belts with segmentation masks using CNNs. They trained and evaluated a Mask R-CNN [6, 10] model for instance segmentation of round fish, such as haddock, cod, whiting, and hake from EM images. Generally, Mask R-CNN is popular in the field and has been used for instance segmentation of fish from EM images [33], in trawls [7], in boxes [24], and in dedicated conveyor belt monitoring systems [23]. The latter based their findings on data captured using the iObserver camera system [36] and illustrated the feasibility of using MobileNetV1 [13] for length estimation of fish.

Recent work on length estimation of fish can be divided into two main approaches. The first one being learning-based methods, such as training a small CNN to regress the length of each fish [23]. The mapping between pixels to real-world lengths, i.e. centimeters, is then indirectly learned by the model. The other group relies on classical image processing, where pixel-wise lengths are first estimated and then afterwards mapped to real-world lengths. An example of extracting the pixel-wise length is to extract the central line of the masks for each fish [27] or by identifying key points based on the convex hull of the fish [29]. The final mapping from pixels to centimeters can then be achieved through depth sensors, such as stereo vision [29] or time-of-flight [27]. Another approach to infer the pixels to centimeters mapping is to rely on a reference object with a known size being present in the image, such as an ArUco marker [21].

Verifying existing findings and driving further development in automated catch monitoring is challenging due to data and annotations typically being kept private. Additionally, conducting fine-grained analysis is often constrained by the targeted species belonging to distinct families, varying significantly in size, or being positioned in a predictable manner. Especially the lack of public datasets is a constraint that is becoming increasingly evident with the growing reliance on data-intensive deep learning models. The few available datasets curated for computer vision tasks are FishNet [14], Fish Detection (FD) [35], Fish Detection and Weight Estimation (FDWE) [30], and DeepFish [8], as outlined in Tab. 1 along with our proposed *AutoFish* dataset.

Dataset name	Images	Labeling type	Object instances	Instances per image*	Location	Environment	IDs	Length	Weight	Both sides
FD [35]	5,231	Bounding boxes	24,008	0-30 (4)	North Sea	Conveyor belt	Yes[1]	No	No	No
FDWE [30]	1,086	Bounding boxes	2,216	1-7 (2)	North Sea	Conveyor belt	No	No	Yes	No
Fishnet [14]	143,818	Bounding boxes	549,209	1-33 (4)[2]	Western and Central Pacific	Vessel deck	No	No	No	No
DeepFish [8]	1,320	Instance segmentation	7,339	1-29 (7)	Spain	Tray	No	Yes	No	No
AutoFish	1,500	Instance segmentation	18,160	7-24 (12)	North Sea	Conveyor belt	Yes	Yes	No	Yes

Table 1. Overview of publicly available and computer vision curated fish catch datasets. **The number in parentheses is the average number of instances per image.* [1]*The IDs in the FD dataset were specifically for the purpose of tracking the fish. All reappearances of each fish correspond to the same sequence with no variation in surroundings or orientation.* [2]*Including humans.*

The FishNet dataset [14] contains images captured from EM cameras on longline tuna vessels in the Pacific. The images are bounding box annotated and contain a single to a few fish. Object resolution is generally low due to a wide field of view, which is typical for EM cameras. The FD [35] and FDWE [30] datasets consist of images captured by a dedicated light and camera system mounted above a conveyor belt. The images are bounding box annotated and contain information regarding occlusion level, while FDWE also includes the weight of the fish. The DeepFish dataset [8] contains images of fish on a tray acquired with an iPhone. The annotations include instance segmentation masks and lengths based on a calibration procedure using the dimensions of the tray on which the fish are placed.

As highlighted in the table, *AutoFish* addresses key gaps within the field, including the lack of instance segmentation and length estimation datasets from conveyor belt environments. Additionally, we include IDs and images of both sides of the fish to support more fine-grained analysis.

3. Dataset

We introduce *AutoFish*, a novel and meticulously curated image dataset for fine-grained analysis of fish on a conveyor belt. The dataset comprises 1,500 high-quality images featuring 454 unique fish with IDs, manual length measurements, and a total of 18,160 instance segmentation masks. It is to our knowledge the only publicly available dataset of catch that includes multiple documented reappearances of the same fish IDs in different orientations and occlusion levels. In this section, we provide a detailed account of the process involved in creating the *AutoFish* dataset.

3.1. Fish composition

Fish used in the AutoFish dataset were caught and landed by a typically Danish commercial fishing vessel conducting trawl fishery in the North Sea, Skagerrak and Kattegat. The samples mainly consisted of fish species with similar visual characteristics including cod (*Gadus morhua*), haddock (*Melanogrammus aeglefinus*), and whiting (*Merlangius merlangus*). These species are all taxonomic members of the cod family, formally named *Gadidae*. Additionally, hake (*Merluccius merluccius*), and horse mackerel (*Trachurus trachurus*) are well represented in the dataset.

All species are of commercial importance and commonly caught in fisheries conducted in the aforementioned areas. Therefore, they are essential for developing a dataset that accurately represents the local industry. An overview of the fish species and the number of individuals included in the *AutoFish* dataset is presented in Figure 2. Note that species represented by only a few individuals are grouped in the *other* category.

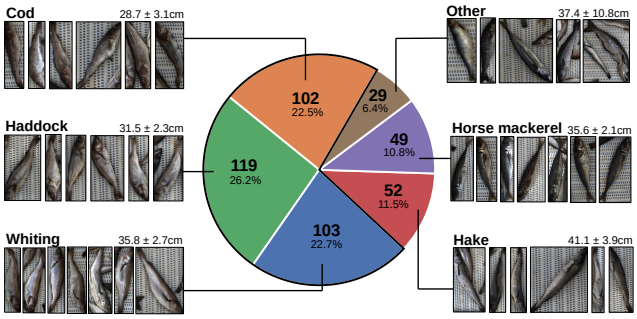

Figure 2. The distribution of species in the AutoFish dataset. The members of the true cod family are highlighted with a black border. The numbers inside the chart indicate the number of specimens. The average length is indicated for each of the species above the image examples.

3.2. Camera setup

The dataset was recorded in a laboratory using a custom setup consisting of a 100x100 cm section of a static white conveyor belt with a camera mounted above, as illustrated in Figure 1. This is similar to the conveyor belt setup that can be commonly found on fishing vessels. We used a Jai GO-5100C-USB camera, equipped with a KOWA LM12HC lens, and it was placed 1.5 m above the conveyor belt. The f-stop and focus distance were set to f/11 and 1 m, respectively, to ensure sharp details across the entire image. The camera was positioned such that the field of view matched the conveyor belt. The images were recorded in RGB with a resolution of 2464×2056 pixels.

Furthermore, camera calibration was carried out by capturing 20 calibration images for each group. Each of the images contains a checkerboard with known dimensions (each square is 20.0 x 20.0 mm) in various poses, including placing the checkerboard flatly on the conveyor belt.

3.3. Image collection

Prior to capturing the images for the dataset, the length of every fish was measured by an experienced marine biologist, following common practice where the number is rounded to the nearest 5 mm. Next, the fish were partitioned into 25 groups, with 14 to 24 fish in each group. As opposed to [8], where the fish appear sorted according to species on the tray, we selected the number of fish and distribution of species in each group pseudo-randomly to mimic real-world scenarios where the fish are processed together when hauled onto the fishing vessel.

The groupings and fish IDs allow us to capture multiple images of the same fish in different orientations and positions, while systematically ensuring that every fish is represented by the same number of images. Additionally, the groups make it convenient to create training and test splits without data cross-contamination.

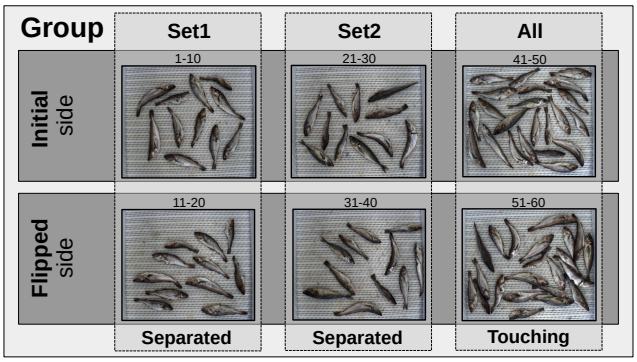

Figure 3. The AutoFish dataset contains 25 groups of fish. Each group consists of three subsets of images, namely, *Set1* and *Set2*, which contain one half of the fish each, and *All*, which contains all of the group's fish.

Every group is partitioned into three subsets: *Set1*, *Set2*, and *All*, each of which comprises 20 images, as illustrated in Fig. 3. *Set1* and *Set2* contain half of the fish each, and none of the fish can overlap or touch each other. On the other hand, *All* contains all the fish in the group, purposely placed in positions where they touch and occlude each other. In the total 60 images of a group, every fish will appear exactly 40 times, with 20 times from each side. This gives us varying levels of difficulties with respect to detection, segmentation, length estimation, and other downstream tasks.

3.4. Annotation procedure

During recording, every fish is meticulously point-annotated with its ID in every image, before shuffling them on the table and repeating the procedure. This allows us to keep track of all fish throughout the session. We provide accurate instance segmentation masks for individual fish in every image of the dataset. We have used the open-source and Python-based annotation software Labelme[1] for annotating the images. First, we leveraged a Segment Anything Model (SAM) [15] to obtain initial segmentation masks based on the manual point-annotations acquired during the image-acquisition procedure. The SAM-based masks were then inspected and manually corrected to ensure the best possible fit. Lastly, in cases where occlusion caused single fish to appear in multiple masks, the masks were associated with the same ID. All annotations were compiled in a single file following the MS COCO [17] format.

4. Methods

Automated catch documentation processes rely on species identification and length estimation. Since fish are non-rigid and can adopt irregular shapes, segmentation masks offer a more precise and visually intuitive method for delineating object boundaries compared to alternatives like bounding boxes or keypoints. This makes masks especially valuable for downstream tasks requiring human inspection and verification. In this work, we generate instance segmentation masks and demonstrate their use in estimating fish lengths through a two-stage process.

4.1. Instance segmentation

In our experiments, we use the instance segmentation model Mask2Former [3] to provide baseline results. Two variations of the architecture are deployed: a configuration equipped with a traditional convolutional ResNet-50 [11] backbone and a larger alternative, using a transformer Swin-base [18] (Swin-B) backbone. All instances of the models are pre-trained on MS COCO [17] and fine-tuned for 1000 steps without a validation set or early stopping. The batch size is set to 8 images. Following the default configuration of Mask2Former, the optimizer used is *ADAMW* [19]. The learning rate is controlled through a multistep scheduler from 0.1 to 0.0001.

During training, we apply common image augmentations and the hyperparameters are outlined in Tab. 2. Random horizontal and vertical flips, with probabilities p_h and p_v, respectively, are applied to make it less likely for the model to learn certain positions and orientations of the fish. As the scene was lit in part by uncontrolled natural light, there is a slight difference between some of the images depending on the time of day, the weather, and more. To minimize the impact of light variation, we introduce contrast (c), brightness (b), and saturation (s) augmentations during training.

4.2. Length estimation

For providing baseline fish length estimations, we implement and evaluate two approaches. The first, denoted *SKL*, relies on classic image processing techniques in the form of applying skeletonization on the instance segmentation masks. The second, denoted *REG*, is a learning-based approach utilizing a small convolutional neural network for length regression.

4.2.1 Mask skeletonization (SKL)

The first step of the skeletonization-based length estimation is to determine the pixel-wise length of each fish along its

Task	p_h	p_v	c	b	s
Segmentation	0.5	0.5	[0.75;1.25]	[0.75;1.25]	[0.75;1.25]
Length est.	0	0	[0.50;1.50]	[0.80;1.20]	[0.60;1.40]

Table 2. Augmentation hyperparameters for the proposed instance segmentation and length estimation models.

[1] https://github.com/labelmeai/labelme

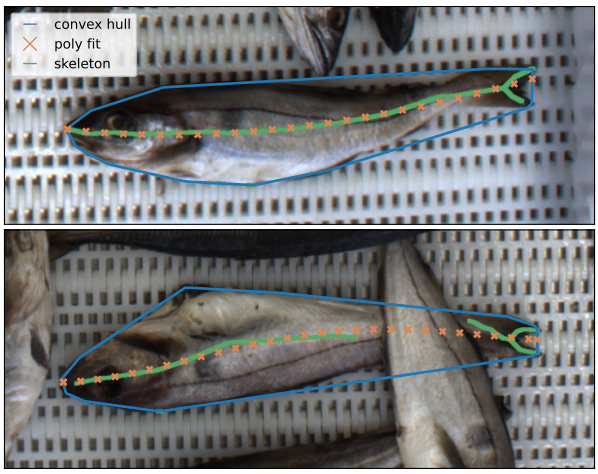

Figure 4. The central line is identified by fitting a polynomial (orange) to the skeleton of the mask (green). Secondly, the polynomial is evaluated based on the convex hull of the mask (blue) to handle forked caudal fins and occlusions.

central line, as illustrated in Fig. 4. We do this by processing the segmentation masks using the skeletonization method proposed by Zhang et al. [38]. To approximate a smooth central line of the fish, from which the length can be estimated, we fit a 4th-degree polynomial to the skeleton of the mask. To account for forked caudal fins and occlusions, where the mask may be split into multiple segments, we compute the convex hull of the mask and evaluate the polynomial along the boundaries of the convex hull.

The next step is to obtain a mapping from image plane (pixels) to the surface of the conveyor belt (centimeters). This is done by estimating a homography for every group of fish based on 20 calibration images, which are also used for correcting lens distortion in the images. Finally, the length in centimeters of each fish can be estimated by taking the image points from the polynomial fit and mapping them onto the plane of the conveyor belt and accumulating the distance between them.

4.2.2 CNN-based length regression (REG)

This approach is inspired by the work of Ovalle et al. [23], which showed that a small MobileNetV1 with a regression head was sufficient for estimating the length of fish on a conveyor belt. We use an ImageNet [4] pre-trained MobileNetV2 [28] model with a custom regression head consisting of two fully connected layers, as shown in Fig. 5.

The masks from the instance segmentation network goes through a pre-processing step before being fed into the network. The input image is cropped to fit a black squared bounding box around the RGB mask of the fish, before it is fed into the MobileNetV2 model. Information regarding the spatial position of the object is critical for the pixel to

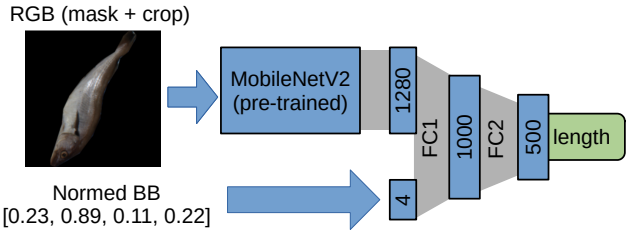

Figure 5. Overview of the CNN-based regression model (REG).

centimeter mapping. Therefore, the normalized bounding box coordinates are provided to the fully connected layers in addition to the embedded image features.

The entire model, including the MobileNetV2 backbone, is trained using a batch size of 32 for a total of 200 epochs. The model is trained using the *L1* loss along with the *ADAM* optimizer using a fixed learning rate of 0.001. During training, randomized image augmentations were used, as reported in Tab. 2. No geometric transformations were applied as they may affect the pixel to centimeter mapping.

5. Results

The *Autofish* dataset is divided into groups to support various split configurations while preventing data cross-contamination between the splits. However, for the evaluation, the following five groups are reserved as the test split: [10, 14, 20, 21, 22] and are excluded from all training. These groups were selected to reflect the overall class distribution of the dataset.

5.1. Instance segmentation

The performance of our instance segmentation models are evaluated based on the mean average precision, which is calculated as $mAP = AP@[IoU = .5 : .95]$ by thresholding the intersection over union (IoU) between the predictions and the ground truth annotations in steps of 0.05.

We evaluate the models per class on the *separated*, *touching*, and *combined* image sets, both training and testing each configuration 10 times with different initializations. The resulting mAPs and error margins are presented in Tab. 3. The performance of both *Mask2Former* models are consistent, with a small error margin of $< 1\%$. Swin-B outperforms the ResNet50 backbone across all species and image categories.

To analyze the impact of the training split size, the models were trained on 20 progressively larger random subsets of the training data. For each subset size (except 20) 10 random variations of groups were used for training the models. The results are presented in Fig. 6, which shows that both models stabilize at 9 groups (~ 180 different fish and ~ 540 images), reaching almost maximum performance with very little variation due to the specific groups used for training.

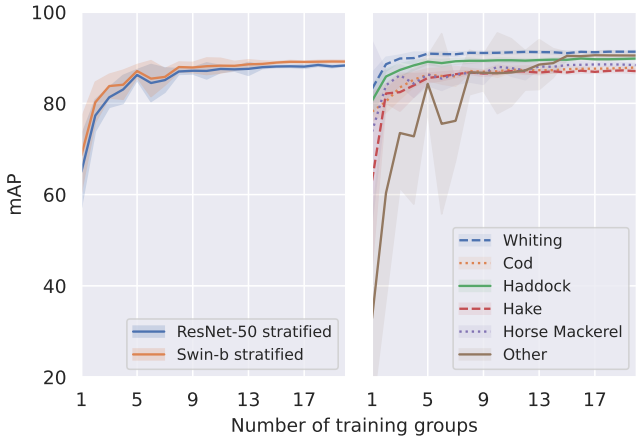

Figure 6. Model mAP and standard deviation as the number of training groups is gradually increased. On the left, a comparison between the two backbones. On the right, the per-class results for the Swin-b backbone.

This indicates the possibility of relatively easy and cheap adaptations to the expected catch of individual vessels.

Characteristic examples of the model's detection capabilities are presented in Fig. 7. Images in the top have a high mAP, close to the general performance of the models, while the images on the bottom feature some of the worst cases. The confidence threshold chosen for the output displayed is 0.9, as our analysis showed that 96% of matched predictions fall above it, with an average mask IoU of 0.94.

5.2. Length estimation

The length estimation methods are evaluated based on the mean absolute error (MAE) in centimeters and the mean absolute percentage error (MAPE). Additionally, the methods are evaluated with two types of input: the Swin-B predictions (denoted *pd*) and the ground truth masks (denoted *gt*). For the *pd* masks, only predictions exceeding a confidence threshold of 90% were used. The training groups were split into a train and a validation split, with 15 and 5 groups, respectively. The validation split was randomly picked to contain the following groups: $[1, 6, 11, 17, 25]$.

(a) mAP = 0.96 (b) mAP = 0.97

(c) mAP = 0.94 (d) mAP = 0.91

(e) mAP = 0.37 (f) mAP = 0.52

Figure 7. Model output examples. Instances of good performance are displayed on the top four images, while low scores are on the bottom. All labels have been predicted correctly.

The performance of the two length estimation approaches are summarized in Table 4, where we see that the skeletonization method (SKL) achieves the best performance on the images from the *separated* sets. However, in the more challenging sets where the fish are allowed to touch and occlude each other, the CNN regression-based

	Separated		Touching		Combined	
	ResNet50	Swin-B	ResNet50	Swin-B	ResNet50	Swin-B
All	92.47 ± 0.12	$\mathbf{92.98 \pm 0.21}$	84.09 ± 0.24	$\mathbf{85.32 \pm 0.32}$	88.31 ± 0.13	$\mathbf{89.15 \pm 0.26}$
Whiting	93.13 ± 0.27	94.20 ± 0.28	87.99 ± 0.37	88.69 ± 0.34	90.47 ± 0.26	91.32 ± 0.27
Cod	91.09 ± 0.13	91.44 ± 0.24	83.27 ± 0.62	83.97 ± 0.47	87.21 ± 0.28	87.86 ± 0.29
Haddock	91.73 ± 0.28	92.63 ± 0.31	85.78 ± 0.24	86.94 ± 0.37	88.75 ± 0.20	89.82 ± 0.26
Hake	90.55 ± 0.35	90.85 ± 0.59	82.18 ± 0.46	83.16 ± 0.38	86.49 ± 0.36	87.06 ± 0.54
Horse mackerel	91.85 ± 0.28	92.29 ± 0.21	82.74 ± 0.82	84.52 ± 0.65	87.29 ± 0.48	88.38 ± 0.28
Other	96.49 ± 0.34	96.50 ± 0.43	82.57 ± 0.93	84.65 ± 0.97	89.66 ± 0.35	90.48 ± 0.54

Table 3. Instance segmentation results for the ResNet50 and Swin-B backbones when trained on all 20 training groups. The mAP and error margin are based on 10 random model initializations.

	Separated	Touching	Combined
SKL^{gt}	**0.59 (1.79%)**	1.43 (4.51%)	1.01 (3.15%)
REG^{gt}	0.67 (2.10%)	**0.96 (3.08%)**	**0.82 (2.59%)**
SKL^{pd}	**0.62 (1.87%)**	2.43 (7.42%)	1.51 (4.59%)
REG^{pd}	**0.62** (1.92%)	**1.38 (4.32%)**	**0.99 (3.10%)**

Table 4. Results for the length estimation reported as MAE in centimeters and MAPE. The skeletonization-based (*SKL*) and regression-based (*REG*) methods are evaluated using both groundtruth masks (*gt*) and actual predicted masks (*pd*).

model (REG) is significantly better.

We present the error distributions of the two approaches in Fig. 8 as histograms, along with the corresponding mean and standard deviation. The notable spikes at the boundaries of the histograms are due to the errors being clipped to a max of ± 5.0 centimeters and hence being accumulated in the outer bins in the histogram. This is done on purpose to avoid a few outliers skewing the plots and also to clearly show the proportion of errors outside this range for the different scenarios.

6. Discussion

The presented baseline experiments show that with state-of-the-art deep neural network architectures, it is possible to automate fish identification on conveyor belts to a large degree. When not presented with heavy occlusion, the models are very consistent at differentiating fish species, even when they look indiscernible to a nonspecialist. In cases with high levels of occlusion, e.g., where fish overlap to the extent where individual fish are divided into multiple segments, the models are able to predict these segments and correctly classify them as belonging to the same individual. It is only in the very extreme cases that the models miss fish entirely. Additionally, the predicted masks are generally of significant quality, with an expected IoU over 0.85. This is critical for the performance of further downstream tasks, like skeletonization, that directly depend on the quality of the mask.

During the evaluation of the length estimation it was found that the skeletonization and CNN-based approaches had a similar performance of 0.62 cm MAE in scenarios with no occlusion (*separated*), as shown in Tab. 4. This suggests that a length estimation approach based on classic image processing, such as skeletonization, is a viable option if there is a low risk of occlusion. This could be valuable for systems mounted on smaller fishing vessels with limited power supply. In the sets where the fish are touching or occluding each other, *touching* and *combined*, the CNN-based approach clearly outperforms the skeletonization-based approach. This is not unexpected, as the skeletonization-based approach is not able to account for occlusions that causes

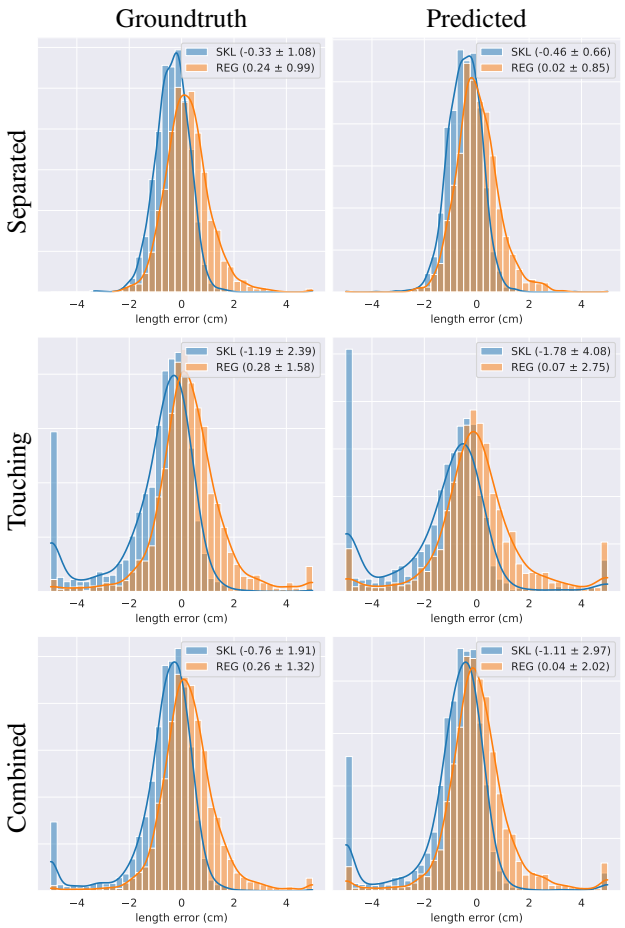

Figure 8. The distribution of length estimation error in centimeters for both the skeletonization-based (*SKL*) and CNN-based (*REG*) approach. All errors have been limited to max ± 5.0 centimeters to avoid outliers skewing the plots. Both the mean error and standard deviation are reported for each approach in the legend of the respective plot.

the head or caudal fin of the fish to be missing from the mask. Therefore, it will have a tendency to underestimate the length of the fish, which is clearly seen in the histograms for the error distribution in Fig. 8. The CNN-based approach, on the other hand, can learn to infer the length based on other features of the mask, in cases where the head or caudal fin are not visible.

A practical distinction between the two length estimation methods is that the skeletonization requires only a few calibration images to adapt to a new setup. In contrast, the CNN necessitates new training data and additional training to be deployed in another environment. To summarize, the skeletonization is easier to adapt to new setups, but struggles with occluded objects. The CNN, on the other hand, handles occlusions effectively but demands new training to adapt to new setups, making it more resource-intensive.

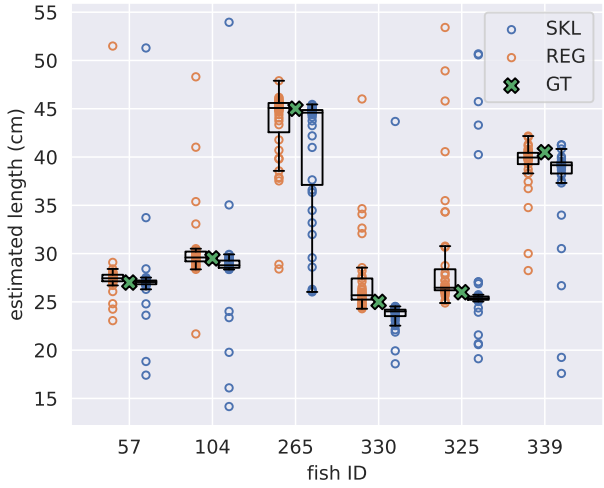

Figure 9. Boxplots for the six fish with the highest variation in their estimated lengths. The actual estimated lengths are plotted for each fish for both the skeletonization-based (*SKL*) and CNN-based (*REG*) approaches. The manually measured ground truth lengths are plotted as well (*gt*).

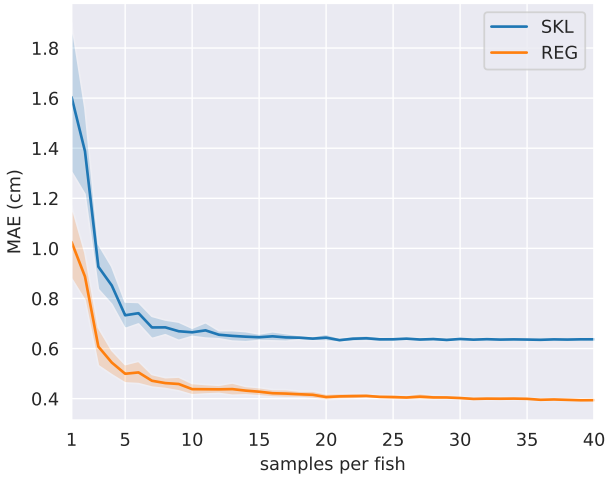

Figure 10. MAE as a function of the number of samples using the median of the length estimation for each fish. The shaded areas denotes the standard deviation of the MAE as it changes depending on which samples are available for the averaging of the lengths. The plot is based on the predicted masks and the *combined* set.

6.1. Actively using IDs in future work

A particular aspect of the *AutoFish* dataset is the inclusion of unique IDs for every fish. This allows for conducting fine-grained analyses of the results, such as the length estimation, on a fish-by-fish level. Individual fish associated with large variations in their estimated lengths were identified and are shown with boxplots in Fig. 9. The samples highlight that there can be a vast difference in the length estimations, even for the CNN-based approach (REG), that is able to handle occlusion to some degree. However, when looking at the median estimated length, both methods are close to the actual ground truth measurement. This suggests that length estimation could benefit from the use of multiple samples for the same fish to increase accuracy. To see how many samples are needed to achieve satisfactory length estimations, the MAE metric was re-calculated using the median estimated length across each individual fish ID with a varying number of samples. The results are plotted in Fig. 10 and simulate a scenario where it is possible to maintain IDs for the individual fish, either through tracking or using re-identification.

Both length estimation approaches appear to benefit from having more samples per fish, but the performance boost is more significant for the CNN-based approach. Identifying 40 samples per fish could potentially boost the MAE from 0.99 cm down to ≈ 0.4 cm for the CNN-based approach. In cases when such a high number of samples per fish is not feasible, having even five samples could still reduce the MAE to ≈ 0.5 cm. This result is at the limit in terms of the precision of the ground truth lengths, as each fish was manually measured to the nearest 5 mm.

Investigating whether it is feasible to maintain fish IDs through re-identification is a promising option for future work with the *AutoFish* dataset. Especially considering that the data are already readily available in the form of 40 images for each of the 454 fish in the dataset. The re-identification task is not only relevant for improving length estimation accuracy, but could also allow for the documentation of fish at an individual level as they are caught or within processing facilities.

7. Ethical statement

Fish used in these experiments were caught and landed by fishermen following relevant legislation and normal fishing procedures. The Danish Ministry of Food, Agriculture and Fisheries of Denmark was contacted before fish collection to ensure compliance with legislation. The fish were dead at landing and only dead fish were included in this experiment. There is no conflict with the European Union (EU) directive on animal experimentation (article 3, 20.10.2010, Official Journal of the European Union L276/39) and Danish law (BEK nr 12, 07/01/2016). The laboratory facilities used at Aalborg University are approved according to relevant legislation.

8. Acknowledgment

The project is financed by the European Union, the European Maritime and Fisheries Fund (EMFF) and the Danish Agricultural and Fisheries Agency (AUTOFISK-33113-I-20-175). We would like to thank Helle Blendstrup, Poul Lund, and Alex Jørgensen for their invaluable help.

References

[1] Luís T. Antelo, Tatiana Ordóñez, Iñaki Miniño, Joaquín Gracia, Emilio Ribes, Juan Hervás, Santiago Simón, and Antonio A. Alonso. A vision-based system for on-board identification and estimation of discarded bio-mass: A tool for contributing to marine resources sustainability. In *OCEANS 2011 IEEE - Spain*, pages 1–8, 2011. 2

[2] Cigdem Beyan and Howard I Browman. Setting the stage for the machine intelligence era in marine science. *ICES Journal of Marine Science*, 77(4):1267–1273, June 2020. 1, 2

[3] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceedings of the 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2022. 4

[4] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2009. 5

[5] Geoffrey French, Mark Fisher, Michal Mackiewicz, and Coby Needle. Convolutional neural networks for counting fish in fisheries surveillance video. In *Procedings of the Machine Vision of Animals and their Behaviour Workshop 2015*. British Machine Vision Association, 2015. 2

[6] Geoff French, Michal Mackiewicz, Mark Fisher, Helen Holah, Rachel Kilburn, Neil Campbell, and Coby Needle. Deep neural networks for analysis of fisheries surveillance video and automated monitoring of fish discards. *ICES Journal of Marine Science*, 77(4):1340–1353, 08 2019. 2

[7] Rafael Garcia, Ricard Prados, Josep Quintana, Alexander Tempelaar, Nuno Gracias, Shale Rosen, Håvard Vågstøl, and Kristoffer Løvall. Automatic segmentation of fish using deep learning with application to fish size measurement. *ICES Journal of Marine Science*, 77(4):1354–1366, Oct. 2019. 2

[8] Nahuel Garcia-d'Urso, Alejandro Galan-Cuenca, Paula Pérez-Sánchez, Pau Climent-Pérez, Andres Fuster-Guillo, Jorge Azorin-Lopez, Marcelo Saval-Calvo, Juan Eduardo Guillén-Nieto, and Gabriel Soler-Capdepón. The deepfish computer vision dataset for fish instance segmentation, classification, and size estimation. *Scientific Data*, 9(1), June 2022. 2, 3

[9] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, June 2014. 2

[10] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. In *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, Oct. 2017. 2

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. 4

[12] Natalie Hold, Lee G. Murray, Julia R. Pantin, Jodie A. Haig, Hilmar Hinz, and Michel J. Kaiser. Video capture of crustacean fisheries data as an alternative to on-board observers. *ICES Journal of Marine Science*, 72(6):1811–1821, Mar. 2015. 1

[13] Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv:1704.04861*, 2017. 2

[14] Justin Kay and Matt Merrifield. The fishnet open images database: A dataset for fish detection and fine-grained categorization in fisheries. *arXiv:2106.09178*, 2021. 2, 3

[15] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 4

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems (NIPS)*, volume 25. Curran Associates, Inc., 2012. 2

[17] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision – ECCV 2014*, page 740–755. Springer International Publishing, 2014. 4

[18] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9992–10002, 2021. 4

[19] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *Proceedings of the 2019 International Conference on Learning Representations (ICLR)*, 2019. 4

[20] Yi-Chin Lu, Chen Tung, and Yan-Fu Kuo. Identifying the species of harvested tuna and billfish using deep convolutional neural networks. *ICES Journal of Marine Science*, 77(4):1318–1329, June 2019. 2

[21] Graham G. Monkman, Kieran Hyder, Michel J. Kaiser, and Franck P. Vidal. Using machine vision to estimate fish length from images using regional convolutional neural networks. *Methods in Ecology and Evolution*, 10(12):2045–2056, 2019. 2

[22] Food & Agriculture Organization of the United Nations (FAO). *The State of World Fisheries and Aquaculture 2020. Sustainability in action.* Food and Agriculture Organization of the United Nations, Rome, Italy, 2020. 1

[23] Juan Carlos Ovalle, Carlos Vilas, and Luís T. Antelo. On the use of deep learning for fish species recognition and quantification on board fishing vessels. *Marine Policy*, 139:105015, May 2022. 2, 5

[24] Miquel Palmer, Amaya Álvarez-Ellacuría, Vicenç Moltó, and Ignacio A. Catalán. Automatic, operational, high-resolution monitoring of fish length and catch numbers from landings using deep learning. *Fisheries Research*, 246:106166, Feb. 2022. 2

[25] Wolfgang Nikolaus Probst. How emerging data technologies can increase trust and transparency in fisheries. *ICES Journal of Marine Science*, 77(4):1286–1294, 03 2019. 1

[26] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 779–788, 2016. 2

[27] Petter Risholm, Ahmed Mohammed, Trine Kirkhus, Sigmund Clausen, Leonid Vasilyev, Ole Folkedal, Øistein Johnsen, Karl Henrik Haugholt, and Jens Thielemann. Automatic length estimation of free-swimming fish using an underwater 3d range-gated camera. *Aquacultural Engineering*, 97:102227, 2022. 2

[28] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4510–4520, 2018. 5

[29] Chen Shi, Qingbin Wang, Xinlei He, Xiaoshuan Zhang, and Daoliang Li. An automatic method of fish length estimation using underwater stereo system based on labview. *Computers and Electronics in Agriculture*, 173:105419, 2020. 2

[30] Maria Sokolova, Manuel Cordova, Henk Nap, Aloysius van Helmond, Michiel Mans, Arjan Vroegop, Angelo Mencarelli, and Gert Kootstra. An integrated end-to-end deep neural network for automated detection of discarded fish species and their weight estimation. *ICES Journal of Marine Science*, 80(7):1911–1922, Aug. 2023. 2, 3

[31] N.J.C. Strachan. Length measurement of fish by computer vision. *Computers and Electronics in Agriculture*, 8(2):93–104, 1993. 2

[32] N.J.C. Strachan. Sea trials of a computer vision based fish species sorting and size grading machine. *Mechatronics*, 4(8):773–783, dec 1994. 2

[33] Chi-Hsuan Tseng and Yan-Fu Kuo. Detecting and counting harvested fish and identifying fish types in electronic monitoring system videos using deep convolutional neural networks. *ICES Journal of Marine Science*, 77(4):1367–1378, May 2020. 2

[34] UnitedNations. Life below water. https://www.un.org/sustainabledevelopment/goal-14-life-below-water/, 2021. Accessed: 2023-02-21. 1

[35] Rick van Essen, Angelo Mencarelli, Aloysius van Helmond, Linh Nguyen, Jurgen Batsleer, Jan-Jaap Poos, and Gert Kootstra. Automatic discard registration in cluttered environments using deep learning and object tracking: class imbalance, occlusion, and a comparison to human review. *ICES Journal of Marine Science*, 78(10):3834–3846, Nov. 2021. 2, 3

[36] C. Vilas, L.T. Antelo, F. Martin-Rodriguez, X. Morales, R.I. Perez-Martin, A.A. Alonso, J. Valeiras, E. Abad, M. Quinzan, and M. Barral-Martinez. Use of computer vision onboard fishing vessels to quantify catches: The iobserver. *Marine Policy*, 116:103714, 2020. 2

[37] D.J. White, C. Svellingen, and N.J.C. Strachan. Automated measurement of species and length of fish by computer vision. *Fisheries Research*, 80(2–3):203–210, Sept. 2006. 2

[38] Tongjie Y Zhang and Ching Y. Suen. A fast parallel algorithm for thinning digital patterns. *Communications of the ACM*, 27(3):236–239, 1984. 5

[39] B Zion, A Shklyar, and I Karplus. Sorting fish by computer vision. *Computers and Electronics in Agriculture*, 23(3):175–187, Sept. 1999. 2