Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/issn/15375110

Research Paper

Automatic measurement of the body length of harvested fish using convolutional neural networks

Chi-Hsuan Tseng [a], Ching-Lu Hsieh [b], Yan-Fu Kuo [a,*]

[a] Department of Biomechatronics Engineering, National Taiwan University, Taipei, Taiwan
[b] Department of Bio-Mechatronics Engineering, National Pingtung University of Science and Technology, Pingtung, Taiwan

ARTICLE INFO

Article history:
Received 14 July 2019
Received in revised form
18 October 2019
Accepted 1 November 2019
Published online 27 November 2019

Keywords:
Convolutional neural networks
Deep learning
Snout-to-fork length
Object detection
Model visualisation

Body lengths of harvested fish are key indices for marine resource management. Some fisheries management organisations require fishing vessels to report the lengths of harvested fish. Conventionally, body lengths of fish are measured manually using rulers or tape measures. Such methods are, however, time consuming, labour intensive, and subjective. This study proposes an automated method to determine the snout-to-fork length of a fish in complex images. In this approach, images of fish bodies and colour plates with a known dimension were acquired. A convolutional neural network (CNN) classifier was then developed to detect the regions of fish head, tail fork, and colour plate in the images. Snout and fork points of the fish were next determined in the fish head and tail fork regions, respectively, using image processing. Fish body length was subsequently estimated as the distance between the snout and fork points using the pixel-to-distance ratio obtained from the colour plate. The developed CNN classifier reached an accuracy of 98.78% in detecting the regions of fish head, fish fork, and colour plate. The proposed approach reached a mean absolute error and a mean absolute relative error of 5.36 cm and 4.26%, respectively, in estimating the body length of fish.

1. Introduction

The body length of harvested fish is essential information. Commercially, the body lengths determine the value of the fish cargoes, and, from the point of view of sustainability, the body length of a fish determines its maturity and is thus used as a control parameter for the management of aquatic resources. Recently, regional fisheries management organisations have required fishing vessels in the offshore fishing industry to report the body lengths of harvested fish (Aranda, de Bruyn, & Murua, 2010; FAO, 2017). Conventionally, the body lengths of the fish are measured manually by observers or fishermen on vessels. However, manual measurement is time consuming and disturbs the work of the fishermen. Moreover, the accuracy of the reported information may be questioned

* *Corresponding author*. Department of Biomechatronics Engineering, National Taiwan University, No. 1, Sec. 4, Roosevelt Rd, Taipei, 106, Taiwan. Fax: +886 2 2362 7620.
E-mail address: ykuo@ntu.edu.tw (Y.-F. Kuo).
https://doi.org/10.1016/j.biosystemseng.2019.11.002

(European Commission, 2015). To expedite the process and improve the accuracy of the information, an automated approach for measuring the body lengths of harvested fish on board the vessels is required.

Image-based approaches have been applied for automatically measuring the body lengths of fish. Strachan (1993) developed a system to estimate the lengths of fish on a transparent conveyor belt by using image processing and fish silhouettes. White, Svellingen, and Strachan (2006) determined the lengths and species of fish for seven species using a conveyor-belt-based machine vision system. Shafry, Rehman, Kumoi, Abdullah, and Saba (2012) evaluated the body lengths of fish from images acquired against a white background using optical theory, geometry concepts, and image processing techniques. Although these approaches achieved automation, they could only be applied to images acquired under specific conditions. Owing to limited space and unpredictable weather, fish images acquired on vessels were usually filled with miscellaneous items in the background and were under uncontrolled illumination (Fig. 1). The aforementioned approaches may be impractical for estimating the body length of fish from these complex images.

Other approaches have sought to estimate length from complex images by relying on manual assistance to identify the locations of fish head, snout, or tail fork. Rochet, Cadiou, and Trenkel (2006) established an underwater video system for assessing the body lengths of swimming fish at depths ranging from 1100 to 1500 m. The system required observers to estimate the length of the fish on screen. Harvey et al. (2003) developed a stereo video system to evaluate the body length of live southern bluefin tuna *in situ*. This system required manual identification of the tuna snouts and tail forks in the images. Hsieh et al. (2011) proposed an algorithm for predicting the length of harvested tuna in the images acquired on the deck of longliners. The algorithm required operators to manually pin out the tuna snouts and forks in the images. Due to the manual aspects of these methods, they are not suitable for fully automatic operation.

To fully automate the measurement of fish body lengths, the most challenging task is to detect and locate the fish heads and tail forks. Convolutional neural networks (CNNs; Krizhevsky, Sutskever, & Hinton, 2012; Simonyan & Zisserman, 2014), a deep learning approach, have been applied for solving tasks of object detection in complex images. There are several examples of this technique being used in other fields of research. Garcia & Delakis, 2004 detected faces of various sizes with rotation up to 20° in complex images using a CNN. Szarvas, Yoshizawa, Yamamoto, and Ogata (2005) recognised pedestrians in arbitrary poses and actions in an unconstrained natural environment using a CNN. Chen, Xiang, Liu, and Pan (2014) located vehicles in high-resolution satellite images using a hybrid deep CNN with sliding window technique. Applications of CNNs related to fish include Jäger, Wolff, Fricke-Neuderth, Mothes, and Denzler (2017) who tracked the movements of multiple fish in underwater video streams using a two-stage graph-based approach and a CNN appearance model. Li, Shang, Qin, and Chen (2015) detected the presence of fish and identified fish species in underwater videos using fast region-based CNN (Girshick, 2015). Another study located fish in coral reef videos and identified fish species using deep CNN models (Zhuang, Xing, Liu, Guo, & Qiao, 2017).

This study aimed to develop a fully automatic approach for measuring the snout-to-fork lengths of harvested fish in complex images. In the proposed approach, images of fish bodies and colour plates (with known dimension of 25×25 cm^2) placed beside were acquired (Fig. 2a). A CNN classifier (Fig. 2b) was then developed to automatically detect the locations of the regions of fish head, tail fork, and colour plate (Fig. 2c). The snout and fork points of the fish and the corners of the colour plates were then determined using image processing (Fig. 2d). The length of the fish was

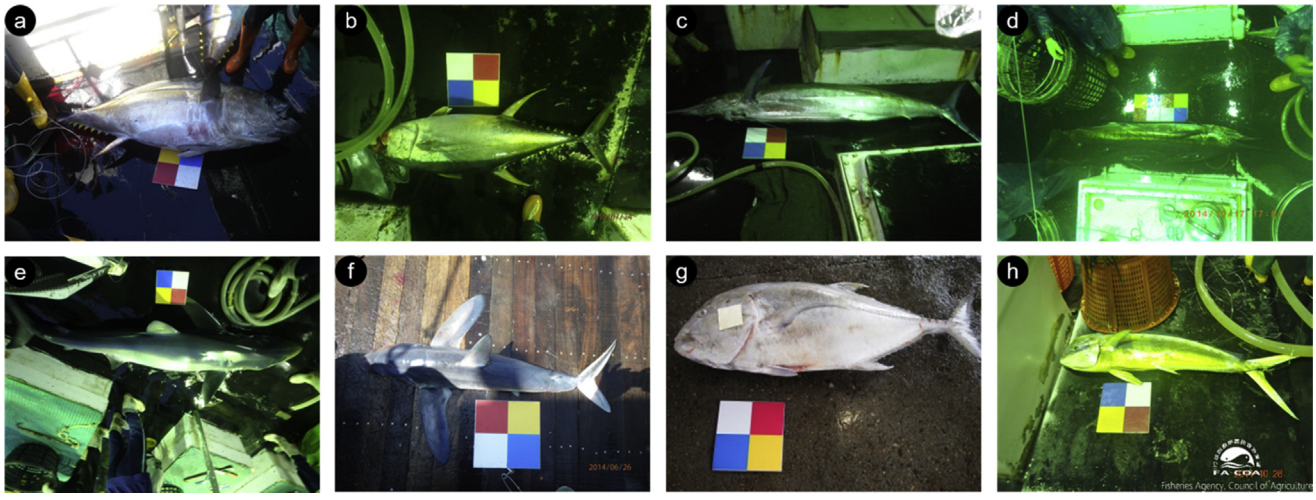

Fig. 1 – Images of common sea fish acquired on the decks of longliners or on the floor of harbours: (a) bigeye tuna (*Thunnus obesus*), (b) yellowfin tuna (*Thunnus albacares*), (c) blue marlin (*Makaira nigricans*), (d) Indo-pacific sailfish (*Istiophorus platypterus*), (e) blue shark (*Prionace glauca*), (f) great white shark (*Carcharodon carcharias*), (g) giant trevally (*Caranx ignobilis*), and (h) common dolphinfish (*Coryphaena hippurus*). Miscellaneous items were randomly distributed on the decks of the longliners when the images were acquired.

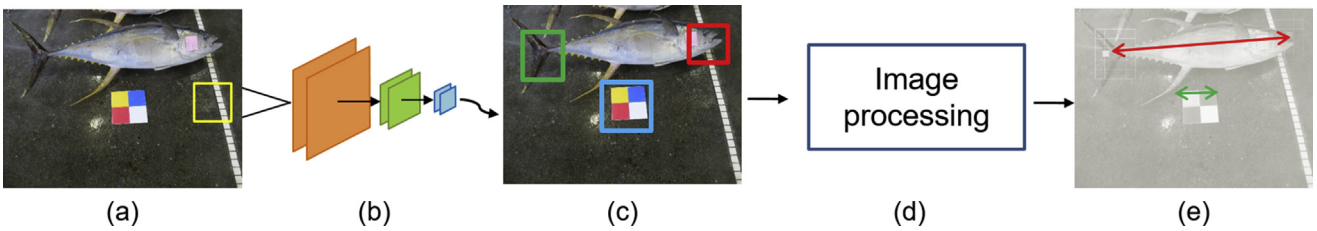

Fig. 2 – The pipeline of fully automatic snout-to-fork fish length measurement: (a) loaded image, (b) CNN classifier, (c) detected regions of fish head, tail fork, and colour plate, (d) image processing, and (e) fish body length estimation.

subsequently estimated using the distance between the snout and fork points and the length-to-pixel ratio obtained from the colour plate (Fig. 2e). The developed CNN classifier was verified using an independent dataset of 4000 images. The proposed length estimation approach was evaluated using an independent set of 154 images.

2. Materials and methods

2.1. Image acquisition and preparation of training patches

A dataset of 9000 fish images was provided by Fisheries Agency, Council of Agriculture (Taiwan). The images had been acquired on the decks of longliners by observers between 2006 and 2017. Five thousand images were used for developing the CNN classifier; the remaining four thousand images were used for assessing the performance of the developed CNN classifier. Another dataset of 154 fish images was acquired at Nan-Fang-Ao fishing harbour (Yilan, Taiwan). The body lengths of the fish in the images in this dataset were manually measured using tape measures when the images were acquired. The body length information was used as validation data to assess the accuracy of the proposed algorithms. The two datasets included various common deep-sea fish (e.g., tuna, billfish, and shark; Fig. 1). The images were acquired under uncontrolled illumination (e.g., sunny days and dark nights) with various backgrounds.

Image patches were cropped from the training images to develop the CNN classifier. A total of 30,000 patches were created: 4000 fish heads, 4000 tail forks, 4000 fish bodies, 4000 colour plates, and 14,000 backgrounds (Fig. 3). Each class of the patches contained characteristics and features of the class. A fish head patch contained the eye, mandible, and upper jaw (Fig. 3a). A tail fork patch contained the tail fin and caudal peduncle (Fig. 3b). A colour plate patch contained a four-colour plate (Fig. 3d). Background patches contained miscellaneous items that appeared at the harbour or on the deck (e.g., shoes, foot parts, floor, and fishing equipment; Fig. 3e). The patches were resized to 36 × 36 pixels. Image augmentation, including rotation, horizontal and vertical shifting, horizontal and vertical flipping, and scaling, was applied to the patches for enhancing the classifier performance. The training samples for developing the CNN classifier comprised 90% of the patches. The remaining patches were used as validation samples.

2.2. Architecture of the proposed CNN classifier

A CNN classifier was developed to distinguish the patches of the five classes. The inputs to the CNN classifier were image patches of 36 × 36 pixels. The architecture of the CNN classifier was adapted from the structures proposed by LeCun, Bottou, Bengio, and Haffner (1998) and Krizhevsky et al. (2012). The classifier consisted of twelve main layers, including six convolutional layers C_1 to C_6, three max pooling layers S_1, S_2, and S_3, and three fully connected layers FC_1, FC_2, and FC_3 (Fig. 4). The max pooling layers S_1, S_2, and S_3 followed the convolutional layers C_2, C_4, and C_6, respectively.

Layers from C_1 to S_3 formed a perception network for feature extraction. The layers contained trainable filters and feature maps. Layers C_1 to C_6, contained 64, 64, 128, 128, 256, and 256 trainable filters of 3 × 3 pixels, respectively. The trainable filters were convoluted with the previous layers. The results were first summed up with trainable biases. To form the feature maps, they were then fed into a non-saturating rectified linear unit (ReLU) activation function (Nair & Hinton, 2010) and then into batch normalisation function (Ioffe & Szegedy, 2015). Consequently, layers C_1 to C_6 contained 64, 64, 128, 128, 256, and 256 feature maps of 36 × 36, 36 × 36, 18 × 18, 18 × 18, 9 × 9 and 9 × 9 pixels, respectively. As a result, layers C_1 to C_6, contained 1792 ($9 \times 64 \times 3 + 64$), 36,928 ($9 \times 64 \times 64 + 64$), 73,856 ($9 \times 128 \times 64 + 128$), 147,584 ($9 \times 128 \times 128 + 128$), 295,168 ($9 \times 256 \times 128 + 256$), and 590,080 ($9 \times 256 \times 256 + 256$) trainable parameters, respectively. Layers S_1, S_2, and S_3 contained 64, 128, and 256 feature maps of 18 × 18, 9 × 9, and 4 × 4 pixels, respectively. These feature maps were results of max pooling (i.e., nonlinear down-sampling) using filters of 2 × 2 pixels with a stride of 2 pixels on the feature maps in the previous layers. There was a total of 1,145,408 trainable parameters in the feature extraction.

Layers FC_1, FC_2, and FC_3, also known as dense layers, formed a perception network for classification. Layer FC_1 had 256 neurons that were fully connected to the 4096 (16 × 256) pixels in layer S_3. Layers FC_2 and FC_3, respectively, had 128 and 5 neurons that were fully connected to the neurons in layers FC_1 and FC_2. The FC_1 neurons were the outputs of the pixels in layer S_3 multiplied with a 4096 × 256 matrix followed by trainable biases. The FC_2 and FC_3 neurons were the outputs of the neurons in layers FC_1 and FC_2 respectively, multiplied with a 256 × 128 and a 128 × 5 matrix followed by trainable biases. As a result, layer FC_1 contained 1,048,832 (4096 × 256) trainable weights and 256 trainable biases. Layers FC_2 and FC_3 contained 32,768 (256 × 128) and 1280 (256 × 5) trainable weights, respectively, and 128 and 5 trainable biases. The outputs of

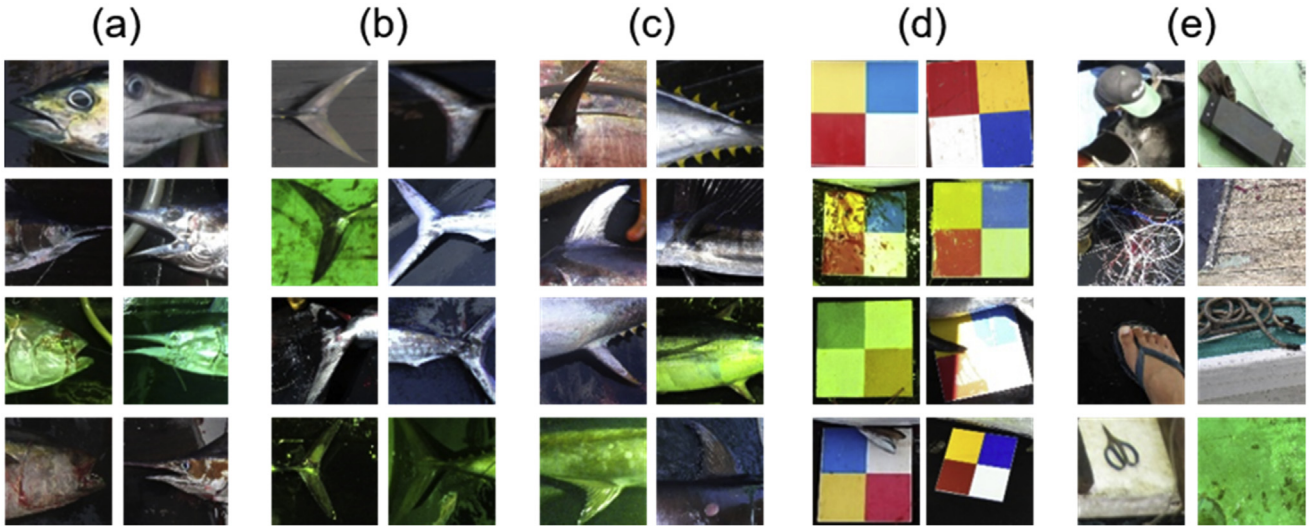

Fig. 3 – Examples of (a) fish head, (b) tail fork, (c) fish body, (d) colour patch, and (e) background patches.

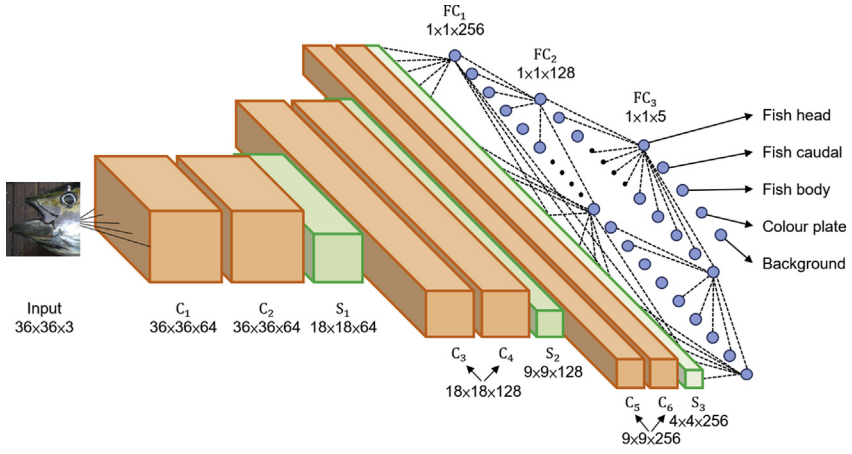

Fig. 4 – Architecture of the proposed CNN classifier.

FC_1 and FC_2 were fed into the ReLU activation function. The outputs of the five FC_3 neurons were fed into softmax activation functions (Bishop, 1995) to represent the probabilities of the five classes, respectively. The proposed CNN classifier contained a total of 2,227,781 trainable parameters. The architecture of the proposed CNN classifier was simpler than some of the well-known deep CNN classifiers, such as AlexNet (Krizhevsky et al., 2012) and VGG-16 (Simonyan et al., 2014).

2.3. *Training of the CNN classifier*

The trainable weights of the CNN classifier were randomly initialised using the Glorot uniform initialiser (Glorot & Bengio, 2010). The trainable biases were initialised as 0. The CNN classifier was trained using backpropagation (LeCun et al., 1990) for 100 epochs. At each epoch, the training samples were shuffled and arranged into 195 batches with a batch size of 128. The classifier weights were optimised using adaptive moment estimation (Kingma & Ba, 2014). The initial learning rate was set to 0.001. Dropout (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014) with a rate of 0.25 was applied to layers S_1, S_2, S_3, FC_1, and FC_2 to prevent overfitting. The CNN classifier was trained using an open-source python environment (Van Rossum & Drake, 1995, p. 130) and a deep learning library Keras (Chollet, 2015) with Tensorflow (Abadi et al., 2016) as the backend. A graphic processing unit (GeForce GTX 1080, Nvidia; Santa Clara, CA) was used to expedite the development of the CNN classifier.

2.4. *Filters visualisation*

The filters of the developed CNN classifier were visualised to realise the information the classifier had gathered (Zeiler & Fergus, 2014). The visualisation was performed following the procedure proposed by Simonyan, Vedaldi, & Zisserman (2013). To visualise a filter, an image with random pixel values was initialised. The image was then fed into the developed classifier, and the response of the filter was calculated. Backpropagation was next applied to update the input image so that the response of the filter was maximised. The aforementioned procedure was performed for 100 iterations. The resultant image was the visualisation of the filter.

2.5. Saliency maps and Grad-CAMs of the developed CNN classifier

Saliency maps (Simonyan et al., 2013) and gradient-weighted class activation maps (Grad-CAMs; Selvaraju et al., 2016) of the developed CNN classifier were generated. A saliency map indicates the importance of each pixel in an input image (Simonyan et al., 2013). In the procedure of generating a saliency map, an input image with a known class was fed into the developed CNN classifier. For each pixel in the input image, the weights of the classifier with respect to the class of the image were calculated using guided backpropagation (Springenberg, Dosovitskiy, Brox, & Riedmiller, 2014). The summation of the weights for each pixel was obtained. The summations arranged as the dimension of the input image then formed the saliency map. A Grad-CAM indicates the importance of each pixel in the feature maps of a classifier (Selvaraju et al., 2016). In the generation of a Grad-CAM, an input image with a known class was fed into the developed CNN classifier to generate the feature maps of the last convolutional layer. Neuron importance was next calculated as the summed gradients of the input image scores with respect to the feature maps. The weighted combination of the feature maps using the neuron importance as the weights was then fed into the ReLU function to form the Grad-CAM of the input image. Guided backpropagation was used to generate the gradients of the input image scores.

2.6. Detection of the fish head, tail fork, and colour plate

The locations of fish head, tail fork, and colour plate in an image were detected using the developed CNN classifier with the spatial pyramid technique (Adelson, Anderson, Bergen, Burt, & Ogden, 1984). In the spatial pyramid process, a fish image (Fig. 5a) was resized to a height of 576 pixels while maintaining the original aspect ratio. The image was next resized with spatial pyramid ratios of 0.3, 0.4, and 0.5 (corresponding to three scales of the images in Fig. 5b). The resized images were then partitioned into patches of 36 × 36 pixels with a stride of 9 pixels (i.e., one-fourth of the patch dimension). Next, the patches were fed into the developed CNN classifier for detecting the fish parts and colour plate. Once detected, the patches were labelled as candidate patches (Fig. 5c). The locations of the candidate patches were then projected back to the original image (Fig. 5d). The union of nearby candidate patches of the same label (e.g., fish head, tail fork, or colour plate) was labelled as a candidate region. If multiple candidate regions had the same label, the largest region was used. The three candidate regions were inwardly reduced by 9 pixels and were then segmented from the original image (Fig. 5e).

2.7. Snout-to-fork length estimation

The snout-to-fork length of the fish was estimated using the information in the fish head, fish tail fork, and colour plate regions. In the estimation, a snout point and a fork point were identified in the head and tail fork regions, respectively, as the ends for the length measurement. The snout point was determined as the point a quarter width of the head region away from the head region centre towards the distal along the image horizon (Fig. 5e). The fork point was determined as the point a quarter width of the tail fork region away from the tail fork region centre towards the distal along the image horizon. The snout-to-fork length in pixels was then estimated as the distance between the snout and fork points in the image.

The snout-to-fork length in cm was estimated using the meter-to-pixel ratio calculated from the colour plate

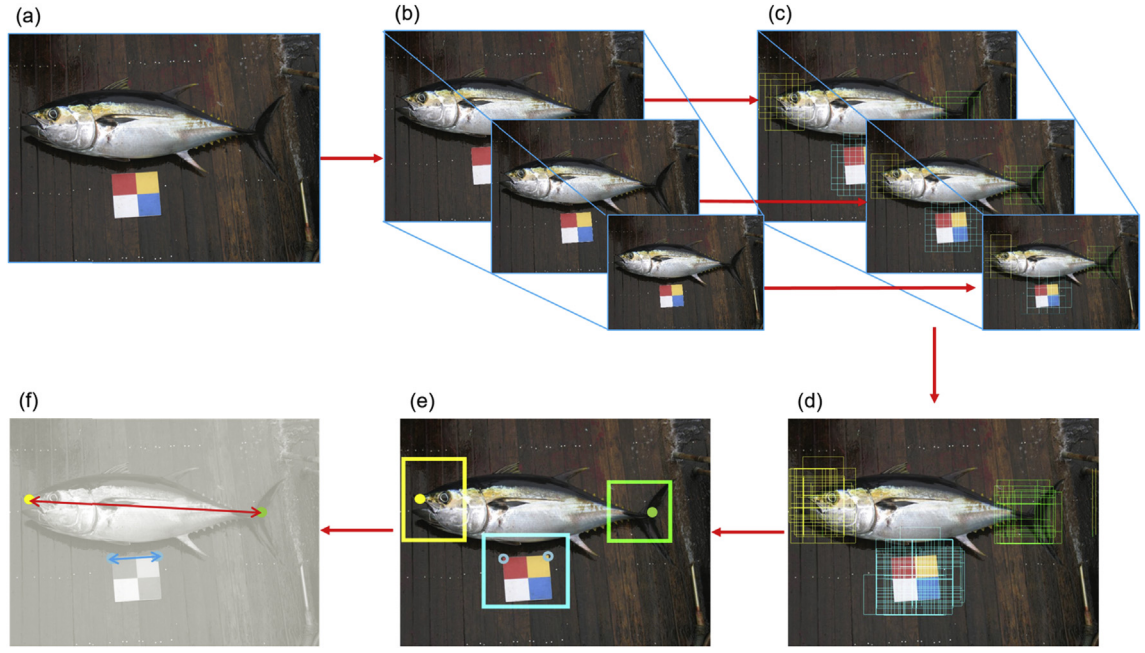

Fig. 5 – Procedure of fish length estimation: (a) original image, (b) image pyramids of the original image, (c) candidate patches of fish head, tail fork, and colour plate, (d) combination of the candidate patches, (e) regions of fish head, tail fork, and colour plate, and (f) snout-to-fork length and side length of the colour plate.

(25×25 cm^2). In the process of the ratio calculation, the colour plate region (Fig. 6a) was converted to grayscale (Fig. 6b). The edges of the colour plate were then detected using a Prewitt operator (Prewitt, 1970, Fig. 6c). The resulting image was then converted to binary using the method introduced by Otsu, 1979 (Fig. 6d). Subsequently, connected component labelling (Haralick & Shapiro, 1992) was applied to the binary image. The largest component in the image was determined as the colour plate. The top-left and top-right corners of the colour plate were detected (Fig. 6e). The side length of the colour plate was then fixed as the distance between the two corners (Fig. 5f) and was used to calculate the meter-to-pixel ratio.

3. Results and discussion

3.1. *Performance of the developed CNN model*

The training accuracy, training loss, validation accuracy, and validation loss during the training of the CNN classifier were examined (Fig. 7). After 40 epochs of training, the validation loss converged to less than 0.13. The validation accuracy reached 97.7% at the end of the training.

The performance of the developed CNN classifier was examined using the 3000 validation patches. The validation accuracies for the fish head, tail fork, and colour plate surpassed 98% (Fig. 8). The body patches only reached an accuracy of 93.2%. However, the proposed method does not rely on the identification of the body patch.

The training time, validation accuracy, and total parameters of the proposed CNN classifier were compared with those of two well-known CNN classifiers, namely VGG-16 and AlexNet (Table 1). The architectures of VGG-16 and AlexNet were adjusted so that their output matched with the five classes used in this work. The training data, procedure, and environments for the AlexNet and VGG-16 classifiers were identical to those of the proposed classifier. Although the architecture of the proposed CNN was much simpler, the results indicated that the proposed CNN model outperformed VGG-16 and AlexNet classifiers in the identification of fish head and tail fork.

3.2. *Filter visualisation of the developed CNN classifier*

The filters in layer FC_3 were visualised (Fig. 9). The filters exhibited clear patterns of the fish head, tail fork, and colour plate. The filter for the fish head exhibited patterns of a fish eye and an upper jaw. The filter for the tail fork exhibited the pattern of a tail fin and caudal peduncle. The filter for the colour plate exhibited the centre and edges of the four colour patches in the plate. However, the filters for fish body and background did not exhibit perceptible features because the images of these two classes usually contained miscellaneous or disorganised information.

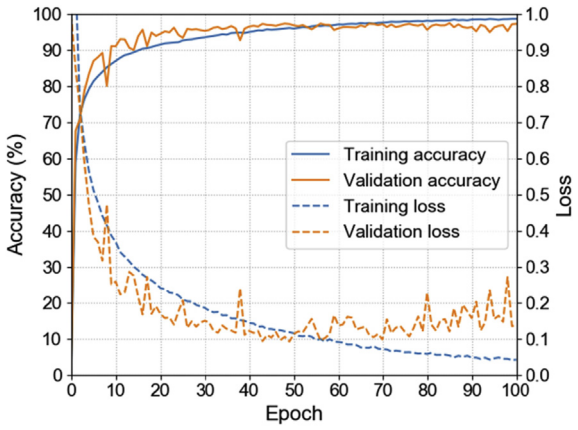

Fig. 7 – Accuracy and loss measured during the training of the CNN classifier.

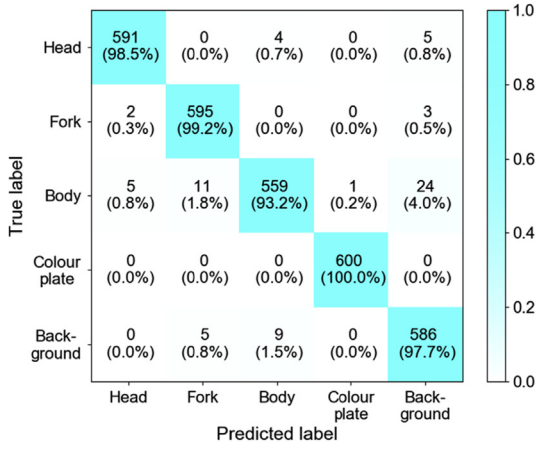

Fig. 8 – Validation accuracy of the developed CNN classifier.

Table 1 – Training time, validation accuracy, and model parameters of the proposed CNN, AlexNet, and VGG-16.

CNN architecture	Training time (s/epoch)	Validation accuracy (%)	Parameters (M)
Proposed	10.0	97.70	2.2
AlexNet	30.0	97.50	37.9
VGG-16	44.0	96.93	65.0

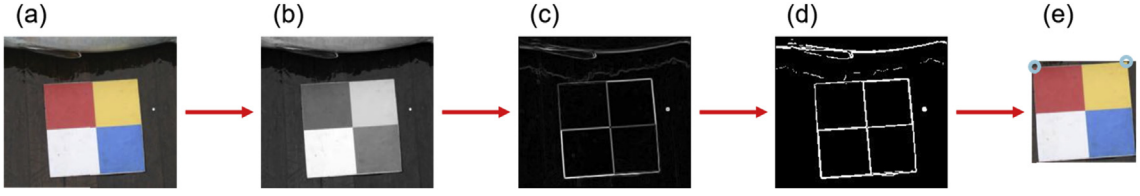

Fig. 6 – Corner detection of a colour plate: (a) colour plate, (b) colour plate in grayscale, (c) edges of the colour plate, (d) edges of the colour plate in binary, and (e) corners of the colour plate.

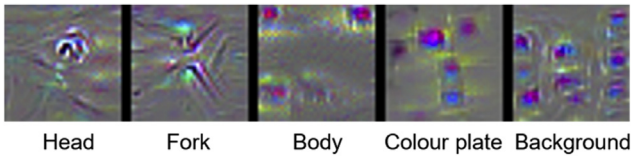

Fig. 9 – Visualisation of the trainable filters in layer FC_3 of the developed CNN classifier.

3.3. *Saliency maps and Grad-CAMs of the developed CNN classifier*

The saliency maps and Grad-CAMs of the developed CNN classifier were generated using input patches of the five classes (Fig. 10). The colours of the pixels in the two maps represented effect of the pixels on classification. In the fish head patches, the pixels around the edges of the head, fish eyes, and fish mouths captured considerable attention. In the tail fork patches, the pixels around the bifurcation gained strong attentions. In the colour plate patches, the pixels around the edges between colour areas and the centre captured considerable attention. The areas where the pixels gained attention were characteristics that human observers pay attention to for identifying the classes.

3.4. *Detection of fish head, tail fork, and colour plate in complex images*

The performance of the developed CNN classifier was assessed using the 4000 test images. Using the spatial pyramid ratios of 0.5, 0.4, and 0.3, the proposed CNN classifier reached an overall accuracy of 98.78% (Table 2). Most of the images were filled with miscellaneous items in the background (Fig. 11). Despite this, the developed CNN classifier identified the regions of fish head and tail fork of various species, including tuna (Fig. 11a, b), billfish (Fig. 11c–f), shark (Fig. 11g), and others (Fig. 11h, i). The illumination conditions of the images varied considerably, including colour shift (Figs. 11d, e), high contrast (Fig. 11b, c, f), and overexposure (Fig. 11a).

Table 2 – Detection rate of fish head, fish tail fork, and colour plate using spatial pyramids.

Spatial pyramid ratio	Head (%)	Fork (%)	Colour plate (%)	All (%)
0.5	89.83	87.37	98.77	78.52
0.4	98.80	96.67	99.20	94.77
0.3	97.14	97.97	98.78	94.58
Combined	99.62	99.23	99.93	98.78

3.5. *Failure case study*

Images with unsuccessful detection of fish head, tail fork, or colour plate were examined (Fig. 12). The failures were caused by inadequate illumination, occlusion, large tilt angle, high contrast in illumination, or combinations of these factors. Figures 12a–c exhibit images with failed detection of fish heads. In these images, the fish heads were tilted at a large angle (Fig. 12a), covered by shadow (Fig. 12b), or obscured with strong colour shifting (Fig. 12c). Figures 12d, e displays images

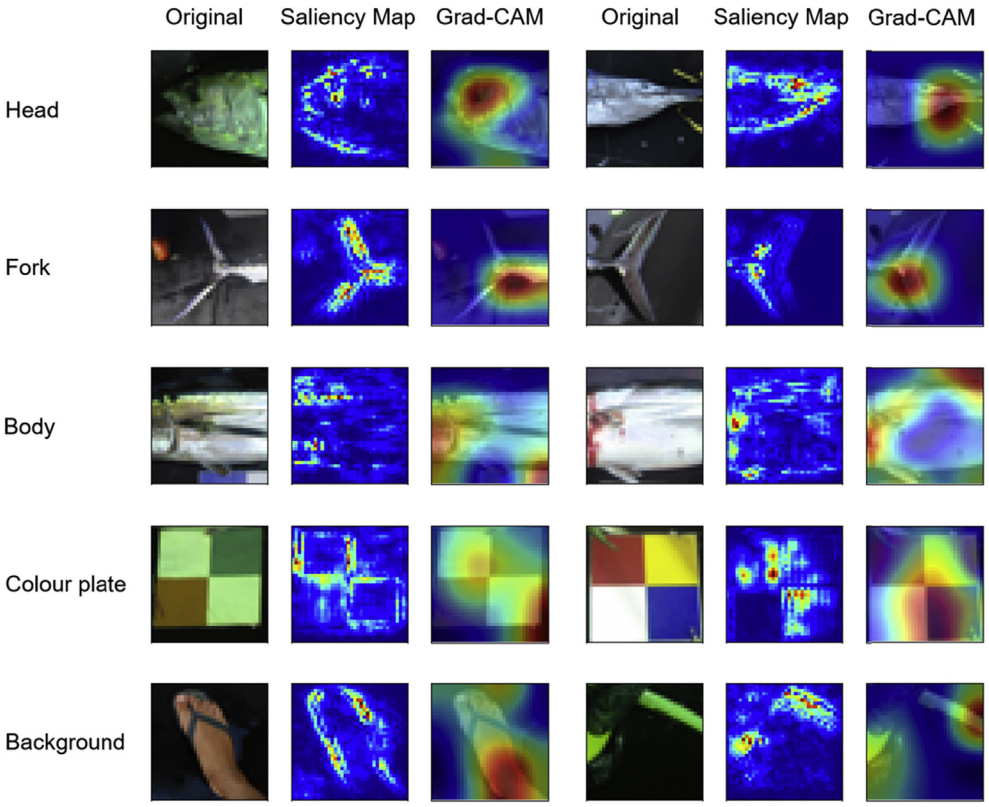

Fig. 10 – Saliency maps and Grad-CAMs of the developed CNN classifier.

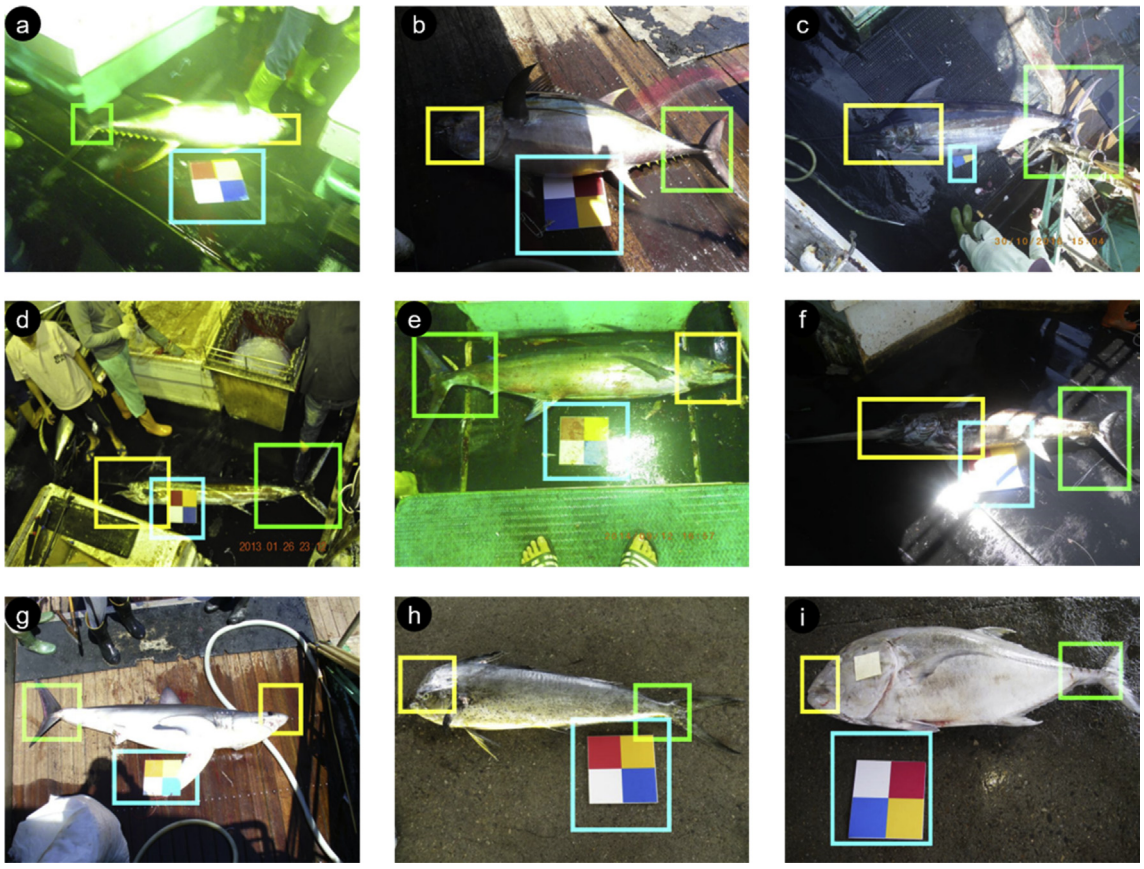

Fig. 11 – Detection of fish head, tail fork, and colour plate in complex images.

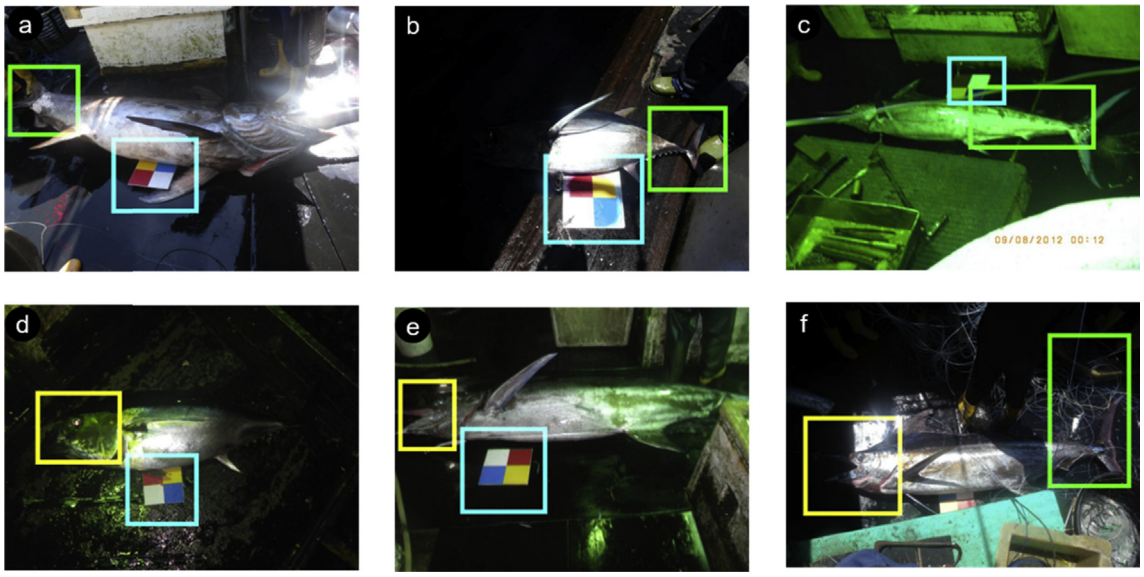

Fig. 12 – Failed detection of (a) fish head due to large tilt angle, (b) fish head due to shadow, (c) fish head due to colour shifting, (d) tail fork due to low illumination, (e) tail fork due to occlusion, and (f) colour plate due to occlusion.

with failed tail fork detection. In these images, the tail forks had limited illumination (Fig. 12d) or were occluded (Fig. 12e). Figure 12f displays an image with failed detection of the colour plate, in which the colour plate was occluded by items on the deck.

3.6. *Detection of the head and tail fork in images of other fish species*

The performance of the developed CNN model was evaluated using fish images of nine species that are not included in

training (Fig. 13). The nine fish species were: (a) long snouted lancetfish (ALX, *Alepisaurus ferox*), (b) great barracuda (BAR, *Sphyraena barracuda*), (c) pomfret (BRZ, *Brama japonica*), (d) common dolphinfish (DOL, *Coryphaena hippurus*), (e) moonfish (LAG, *Lampris guttatus*), (f) escolar (LEC, *Lepidocybium flavobrunneum*), (g) oilfish (OIL, *Ruvettus pretiosus*), (h) ribbonfish (TRP, *Trachipterus ishikawae*), and (i) wahoo (WAH, *Acanthocybium solandri*). Twenty images were used per fish species. The proposed CNN classifier reached an overall detection rate of 88.33% in detecting all the three objects (fish head, tail fork, and colour plate; Table 3). The CNN model reached a high detection rate on ALX, BRZ, DOL, and OIL. These fish species have heads and fork tails that resemble the heads and fork tails of tuna or billfish to a high degree. However, the CNN model only reached a detection rate of 50% on TRP (Fig. 13h). TRP has a small tail fin, resulting in the low detection rate of the tail fork region. The performance of the CNN model may be further improved if the training images include the fish with unique appearances.

3.7. Accuracy of the snout-to-fork length estimation

Two datasets were used for accessing the performance of the proposed method for estimating the snout-to-fork lengths: (a) 100 fish images randomly selected from the 4000 test images and (b) 154 fish images acquired at Nan-Fang-Ao fishing harbour. For the dataset of the 100 images, the snout-to-fork lengths of the fish were manually measured by pinning out the snout and fork points and the corners of the colour plates in the images. The proposed method achieved a mean absolute error (MAE) and a mean absolute relative error (MARE) of 5.58 cm and 4.42%, respectively (Fig. 14a). The coefficient of determination (R^2) of the length estimation was 0.950.

For the dataset of the 154 fish images, the snout-to-fork lengths of the fish were manually measured using tape measures when the images were acquired. The proposed method achieved an MAE and an MARE of 5.36 cm and 4.26%, respectively, on the dataset (Table 4). The R^2 of the length estimation was 0.960. The dataset included 87 tuna, 38 billfish, and 29 other fish. The results indicated that the errors did not significantly differ among fish types (ANOVA; $P = 0.14$; Fig. 14b).

The error of the snout-to-fork length estimation could be contributed by three main factors: (1) inaccurate localisation of fish head and tail fork regions, (2) distortion caused by the cameras used for image acquisition, and (3) manual length measurement error. In certain cases, the regions of fish head and tail fork were not detected accurately by the developed CNN (fish head in Fig. 15a and tail fork in Fig. 15b). In other certain cases, fish body images were acquired with high levels

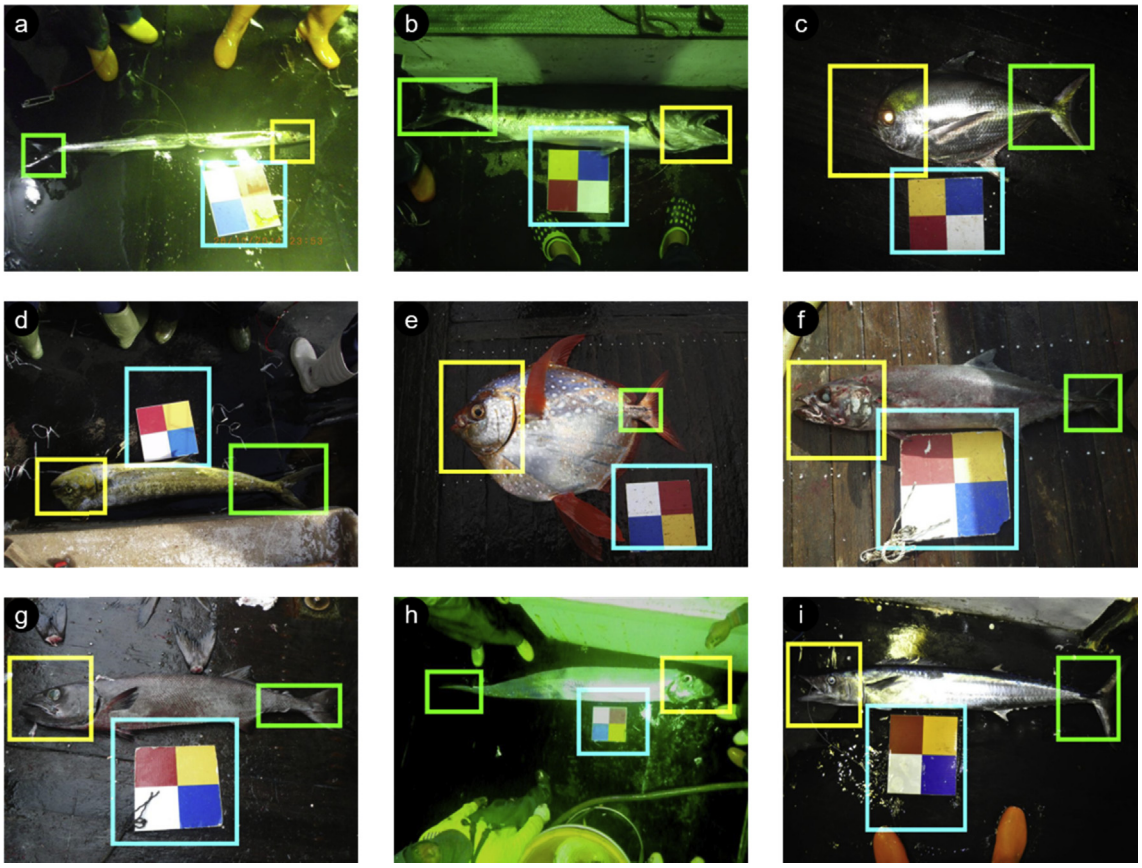

Fig. 13 – Detection of fish head, tail fork, and colour plate on fish species not included in training: (a) long snouted lancetfish (*Alepisaurus ferox*), (b) great barracuda (*Sphyraena barracuda*), (c) pomfret (*Brama japonica*), (d) common dolphinfish (*Coryphaena hippurus*), (e) moonfish (*Lampris guttatus*), (f) escolar (*Lepidocybium flavobrunneum*), (g) oilfish (*Ruvettus pretiosus*), (h) ribbonfish (*Trachipterus ishikawae*), and (i) wahoo (*Acanthocybium solandri*).

Table 3 – Detection rate of fish head, fish tail fork, and colour plate on fish species not included in the training.

Type	ALX	BAR	BRZ	DOL	LAG	LEC	OIL	TRP	WAH	Overall
Image	20	20	20	20	20	20	20	20	20	180
Head (%)	100	100	100	100	100	100	100	80	100	97.77
Fork (%)	100	80	100	100	80	90	100	50	95	88.33
Colour plate (%)	100	100	100	100	100	100	100	100	100	100
All (%)	100	80	100	100	80	90	100	50	95	88.33

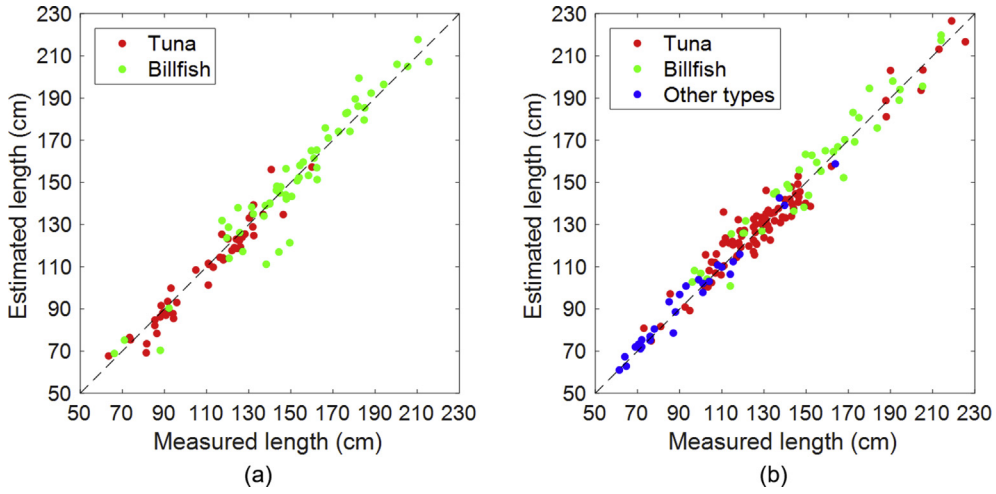

Fig. 14 – Measured and estimated lengths of (a) 100 fish images randomly selected from the test image dataset and (b) 154 fish images acquired at Nan-Fang-Ao fishing harbour.

of perspective distortion owing to limited space for image acquisition. In these cases, the fish and square colour plates were deformed (Fig. 15c, d). The levels of perspective distortion were high particularly for fish with long body lengths. Furthermore, the reference for the snout-to-fork lengths of the fish was manually measured. Manual measurement may introduce error due to fatigue. The aforementioned reasons may result in the MARE of approximately four percent in length estimation.

Table 4 – The error of the snout-to-fork length estimation for the three fish types in the 154 fish dataset.

Class	Tuna	Billfish	Other	Overall
Image	87	38	29	154
MAE (cm)	5.33	7.17	3.08	5.36
MARE (%)	4.24	5.01	3.33	4.26

3.8. *Potential implications of the proposed approach*

The proposed approach could be used on vessels or in harbours for assisting the collection of fish body lengths. In recent years, electronic monitoring systems (EMSs) have been implemented on vessels to record fishing practices. Closed-circuit television (CCTV) cameras are also used in harbours to monitor fish unloading. Currently the length measurement of fish body in EMS or CCTV images is still manual. With appropriate hardware installation (e.g., colour plates in the background), the proposed approach could be used to estimate the lengths of fish in EMS and CCTV images automatically. With the combination of fish species identification (Lu, Tung, & Kuo, 2019), EMS and CCTV systems could possess the abilities to generate the information of harvested fish automatically. Thousands of hours for manually analysing EMS and CCTV images could be saved. CNN models may require high level of computation power. If EMS or CCTV

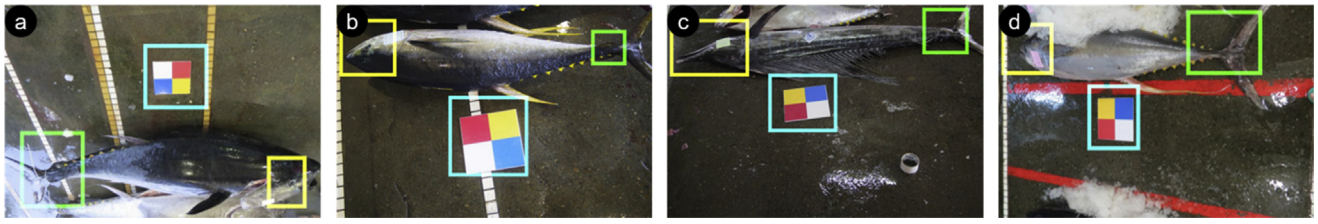

Fig. 15 – Error sources in body length measurement: (a) inaccurate localisation in fish head, (b) inaccurate localisation in tail fork, and (c) (d) high levels of perspective distortion of the acquired images.

systems have limited computation capacities, the analysis of EMS or CCTV images could be conducted at data centres possessing high computation capacities as an alternative.

4. Conclusion

In this study, an automated measurement method was developed for the snout-to-fork lengths of fish in complex images. In this approach, images of fish bodies and colour plates with a known dimension were first acquired. A CNN classifier was developed and applied for detecting the regions of fish head, tail fork, and colour plate in the images. The snout and fork points of the fish were next determined in the fish head and tail fork regions, respectively, using image processing. The lengths of the fish were subsequently estimated as the distances between the two points using the length-to-pixel ratios obtained from the colour plate regions. The filters, saliency maps, and Grad-CAMs of the developed CNN classifier were visualised to demonstrate the feature that the CNN classifier learned from the training images. The developed CNN classifier reached an overall accuracy of 98.78% in detecting the regions of fish head, tail fork, and colour plate. The proposed method reached an MAE and an MARE of 5.36 cm and 4.26%, respectively, in estimating the snout-to-fork length.

Declaration of Competing Interest

None.

Acknowledgments

This research was supported by the Fish Agency, Council of Agriculture, Executive Yuan, Taiwan, under the grants 106AS-18.1.7-FA-F1, 107AS-14.2.7-FA-F1, and 108AS-13.2.7-FA-F1.

REFERENCES

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., et al. (2016). TensorFlow: A system for large-scale machine learning. In , 16. *OSDI* (pp. 265–283).

Adelson, E. H., Anderson, C. H., Bergen, J. R., Burt, P. J., & Ogden, J. M. (1984). Pyramid methods in image processing. *RCA engineer, 29*(6), 33–41.

Aranda, M., de Bruyn, P., & Murua, H. (2010). A report review of the tuna RFMOs: CCSBT, IATTC, IOTC, ICCAT and WCPFC. *EU FP7 Project, 212188*, 171.

Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford university press.

Chen, X., Xiang, S., Liu, C. L., & Pan, C. H. (2014). Vehicle detection in satellite images by hybrid deep convolutional neural networks. *IEEE Geoscience and Remote Sensing Letters, 11*(10), 1797–1801.

Chollet, F. (2015). Keras. https://github.com/fchollet/keras.

European Commission. (2015). Fighting illegal fishing: Commission warns Taiwan and Comoros with yellow cards and welcomes reforms in Ghana and Papua New Guinea. Press Release]. Retrieved from http://europa.eu/rapid/press-release_IP-15-5736_en.htm.

FAO. (2017). *Seafood traceability for fisheries compliance: Country-level support for catch documentation schemes*. Rome: FAO.

Garcia, C., & Delakis, M. (2004). Convolutional face finder: A neural architecture for fast and robust face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 26*(11), 1408–1423.

Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision* (pp. 1440–1448).

Glorot, X., & Bengio, Y. (2010). Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics* (pp. 249–256).

Haralick, R. M., & Shapiro, L. G. (1992). *Computer and Robot Vision. I*. Addison-Wesley.

Harvey, E., Cappo, M., Shortis, M., Robson, S., Buchanan, J., & Speare, P. (2003). The accuracy and precision of underwater measurements of length and maximum body depth of southern bluefin tuna (Thunnus maccoyii) with a stereo–video camera system. *Fisheries Research, 63*(3), 315–326.

Hsieh, C. L., Chang, H. Y., Chen, F. H., Liou, J. H., Chang, S. K., & Lin, T. T. (2011). A simple and effective digital imaging approach for tuna fish length measurement compatible with fishing operations. *Computers and Electronics in Agriculture, 75*(1), 44–51.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning* (pp. 448–456).

Jäger, J., Wolff, V., Fricke-Neuderth, K., Mothes, O., & Denzler, J. (2017). Visual fish tracking: Combining a two-stage graph approach with CNN-features. In *OCEANS 2017-Aberdeen* (pp. 1–6). IEEE.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980. In *Proceedings of the 3rd International conference for learning representations, San Diego, California*.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems* (pp. 1097–1105).

LeCun, Y., Boser, B. E., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W. E., et al. (1990). Handwritten digit recognition with a back-propagation network. In *Advances in neural information processing systems* (pp. 396–404).

LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278–2324.

Li, X., Shang, M., Qin, H., & Chen, L. (2015). Fast accurate fish detection and recognition of underwater images with fast R-CNN. In *OCEANS'15 MTS/IEEE Washington* (pp. 1–5). IEEE.

Lu, Y. C., Tung, C., & Kuo, Y. F. (2019). Identifying the species of harvested tuna and billfish using deep convolutional neural networks. *ICES Journal of Marine Science*. https://doi.org/10.1093/icesjms/fsz089.

Nair, V., & Hinton, G. E. (2010). Rectified linear units improve restricted Boltzmann machines. In *Proceedings of the 27th international conference on machine learning ICML-10* (pp. 807–814).

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics, 9*(1), 62–66.

Prewitt, J. M. (1970). Object enhancement and extraction. *Picture processing and Psychopictorics, 10*(1), 15–19.

Rochet, M. J., Cadiou, J. F., & Trenkel, V. M. (2006). Precision and accuracy of fish length measurements obtained with two visual underwater methods. *Fishery Bulletin, 104*(1), 1–9.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2016). Grad-cam: Visual explanations from deep

networks via gradient-based localization, 3, 7(8) https://arxiv.org/abs/1610.02391.

Shafry, M. R. M., Rehman, A., Kumoi, R., Abdullah, N., & Saba, T. (2012). FiLeDI framework for measuring fish length from digital images. *International Journal of the Physical Sciences, 7*(4), 607–618.

Simonyan, K., Vedaldi, A., & Zisserman, A. (2013). Deep inside convolutional networks: Visualising image classification models and saliency maps. arXiv preprint arXiv:1312.6034. In *Proceedings of the International conference on learning representations*.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556. In *Proc. International conference on learning representations*.

Springenberg, J. T., Dosovitskiy, A., Brox, T., & Riedmiller, M. (2014). Striving for simplicity: The all convolutional net. arXiv preprint arXiv:1412.6806. In *Proc. ICLR, 2015* (pp. 1–14).

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research, 15*(1), 1929–1958.

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

Szarvas, M., Yoshizawa, A., Yamamoto, M., & Ogata, J. (2005). Pedestrian detection with convolutional neural networks. In *Intelligent vehicles symposium, 2005. Proceedings. IEEE* (pp. 224–229). IEEE.

Van Rossum, G., & Drake, F. L., Jr. (1995). *Python tutorial*. Amsterdam, The Netherlands: Centrum voor Wiskunde en Informatica.

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

Zeiler, M. D., & Fergus, R. (2014). Visualizing and understanding convolutional networks. In *European conference on computer vision* (pp. 818–833). Cham: Springer.

Zhuang, P., Xing, L., Liu, Y., Guo, S., & Qiao, Y. (2017). Marine animal detection and recognition with advanced deep learning models. In *Working Notes of CLEF*, 2017.