ICES Journal of
Marine Science

International Council for the Exploration of the Sea
Conseil International pour l'Exploration de la Mer

ICES Journal of Marine Science (2019), doi:10.1093/icesjms/fsz089

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

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

Yi-Chin Lu[1], Chen Tung[1], and Yan-Fu Kuo [1]*

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

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

Lu, Y.-C., Tung, C., and Kuo, Y.-F. Identifying the species of harvested tuna and billfish using deep convolutional neural networks. – ICES Journal of Marine Science, doi:10.1093/icesjms/fsz089.

Received 15 November 2018; revised 21 February 2019; accepted 26 April 2019.

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

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

Introduction

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

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

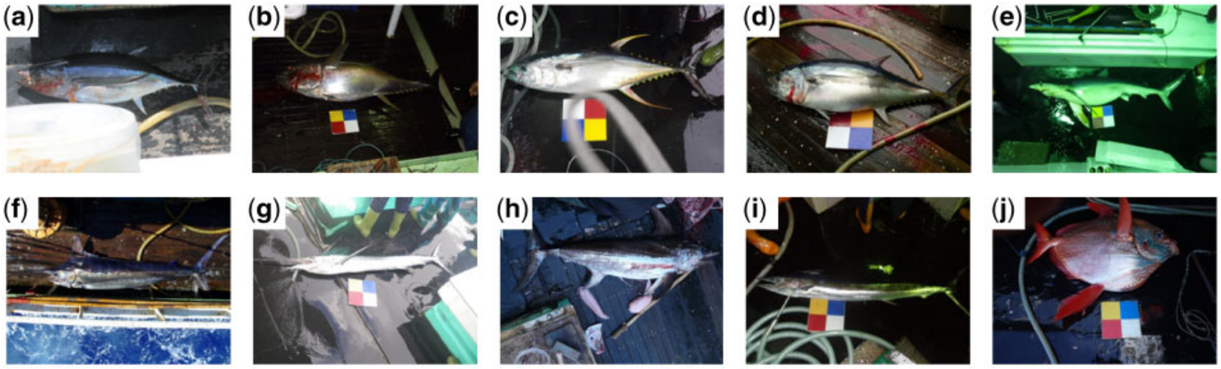


Figure 1. Images of (a) albacore, (b) big eye tuna, (c) yellowfin tuna, (d) other tuna, (e) shark, (f) blue marlin, (g) Indo-pacific sailfish, (h) swordfish, (i) other billfish, and (j) moonfish.

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

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

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

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

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

Material and methods

Image collection

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

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

Image preprocessing, cross-validation, and image augmentation

The dimensions of the fish images ranged from 640 × 360 to 4608 × 3456 pixels. To reduce the complexity of the CNN models, the images were resized to 330 × 250 pixels. Zero padding was applied to the resized images for maintaining the aspect ratio of the images. Subsequently, image augmentation was applied to the images for model training (i.e. training images). Image manipulation generalizes the images and, hence, increases the robustness of the models to be developed. The augmentation operations included horizontal flipping, vertical flipping, width shifting (randomly between −33 and 33 pixels), height shift (randomly between −25 and 25 pixels), rotation (randomly between 0° and 30°), shearing (randomly between 0 and 66 pixels), zoom-in (randomly between 1 and 1.2), and zoom-out (randomly between 0.8 and 1) (Figure 2). Each operation was randomly applied to the images before they were used for training.

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

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

Strategies for fish species identification

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

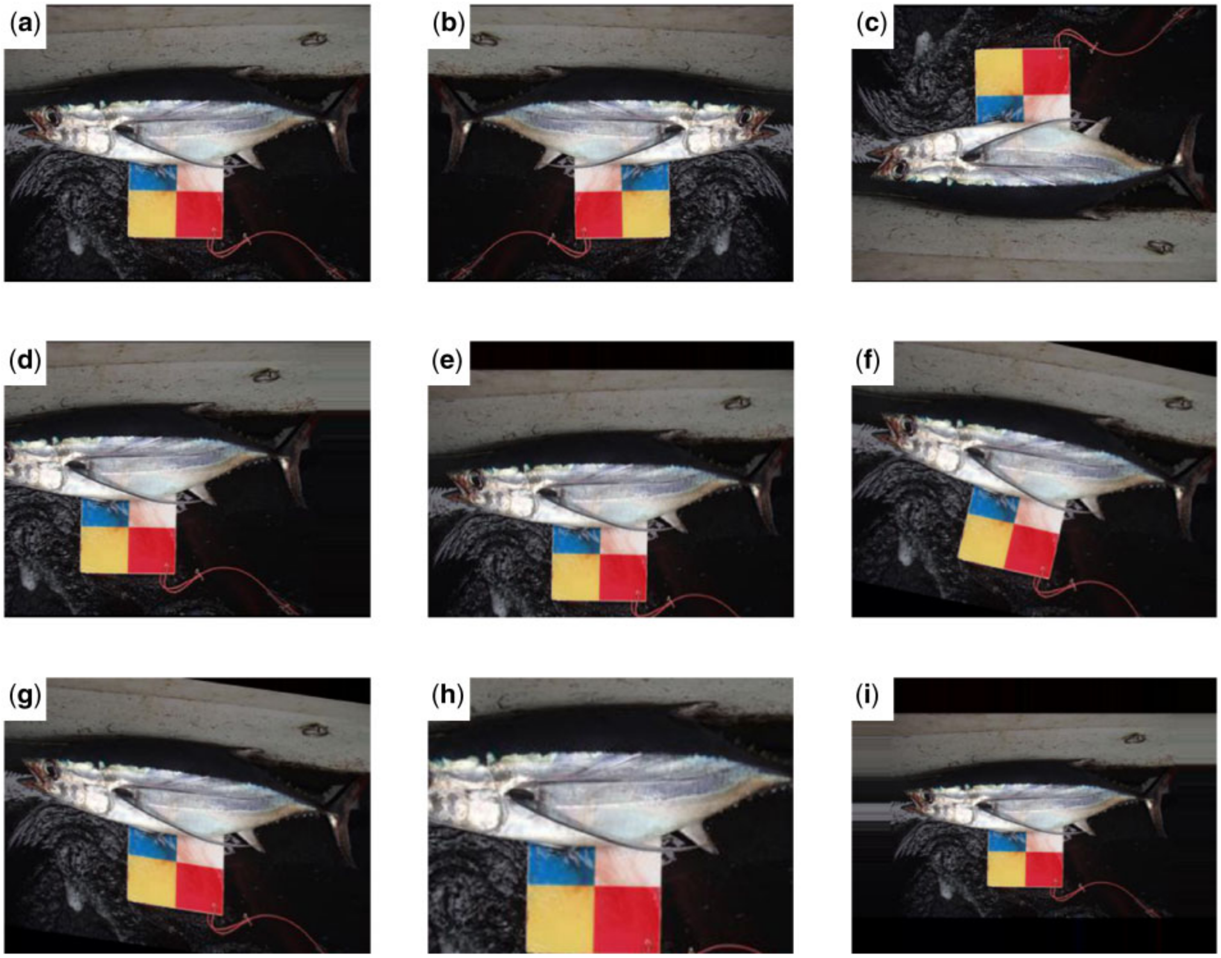


Figure 2. Image manipulation: (a) original image, (b) horizontal flipping, (c) vertical flipping, (d) width shift, (e) height shift, (f) rotation, (g) shearing, (h) zoom-in, and (i) zoom-out.

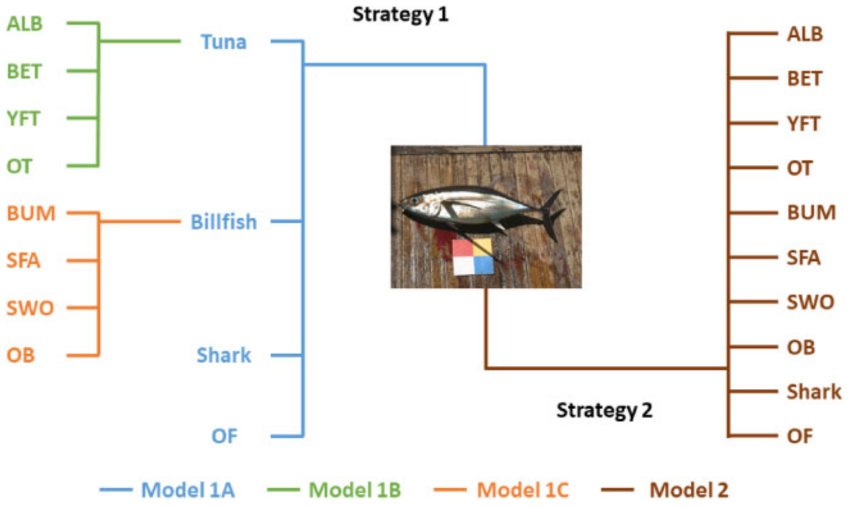


Figure 3. Two strategies for fish type and species identification. Strategy 1 uses three models to identify fish types, tuna species, and billfish species. Strategy 2 uses a single model to identify fish types and species.

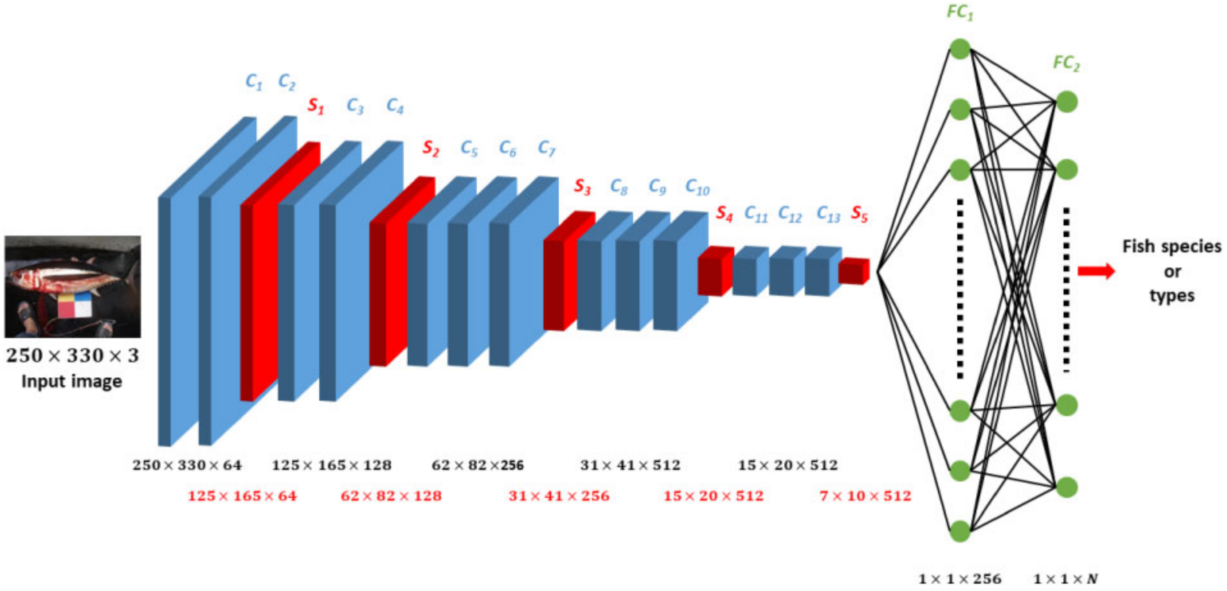


Figure 4. Architecture of the modified VGG-16 model. C: convolution layer, S: max pooling layer, and FC: fully connected layer.

Model development using transfer learning

Transfer learning was applied to the development of deep CNN models. In this procedure, a model with parameters pre-trained using other datasets was adapted. The structures of the output layers were modified to match the output dimensions (i.e. types or species). Next, some layers of the model were frozen. Fine-tuning was then applied to the remaining layers of the model to update the parameters. In this study, VGG-16 (Simonyan and Zisserman, 2014) was chosen as the pre-trained model because the architecture performed well in various classification tasks and was used in numerous applications (Ballas *et al.*, 2015; Liu *et al.*, 2016; Lopez *et al.*, 2017; Abas *et al.*, 2018). Originally, VGG-16 consisted of 13 convolutional (C_1 to C_{13}), 5 max pooling (S_1 to S_5), and 3 fully connected (FC_1 to FC_3) layers (Figure 4). A convolutional layer applies convolution operations to the neurons in the current layer using filters and passes the results to the next layer. A pooling layer combines the neurons in the current layer into a single neuron in the next layer (Huang *et al.*, 2007). A fully connected layer connects every neuron in the current layer to every neuron in the next layer (Viglione, 1970). Convolutional layers C_1 to C_{13} contained 64, 64, 128, 128, 256, 256, 256, 512, 512, 512, 512, 512, and 512 filters, respectively. The dimension and stride of the filters in the convolutional layers were 3 × 3 pixels and 1 pixel, respectively. Zero padding was used in the convolution operations to keep the dimension of the output the same as that of the input. The dimension and stride of the filters in the max pooling layers were 2 × 2 pixels and 2 pixels, respectively.

In this study, the architecture of VGG-16 was adjusted by replacing the original FC layers with new FC layers (FC_1 and FC_2 in Figure 4) with dimension of R^{256} and R^N, where N is the

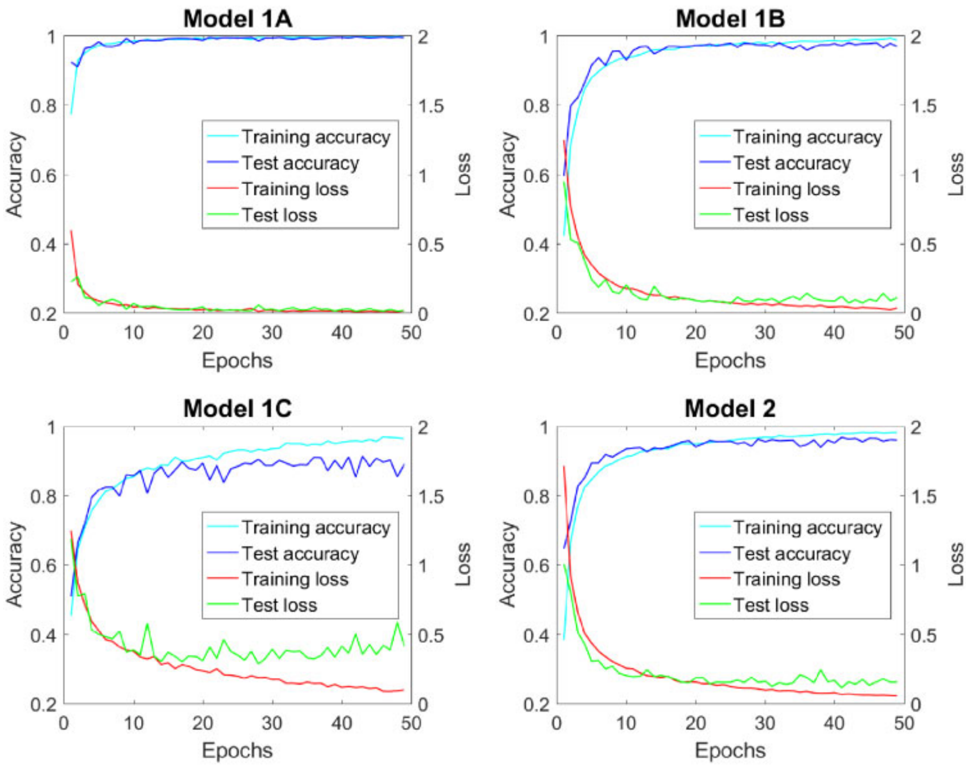


Figure 5. Model accuracy and loss during training.

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

Model training

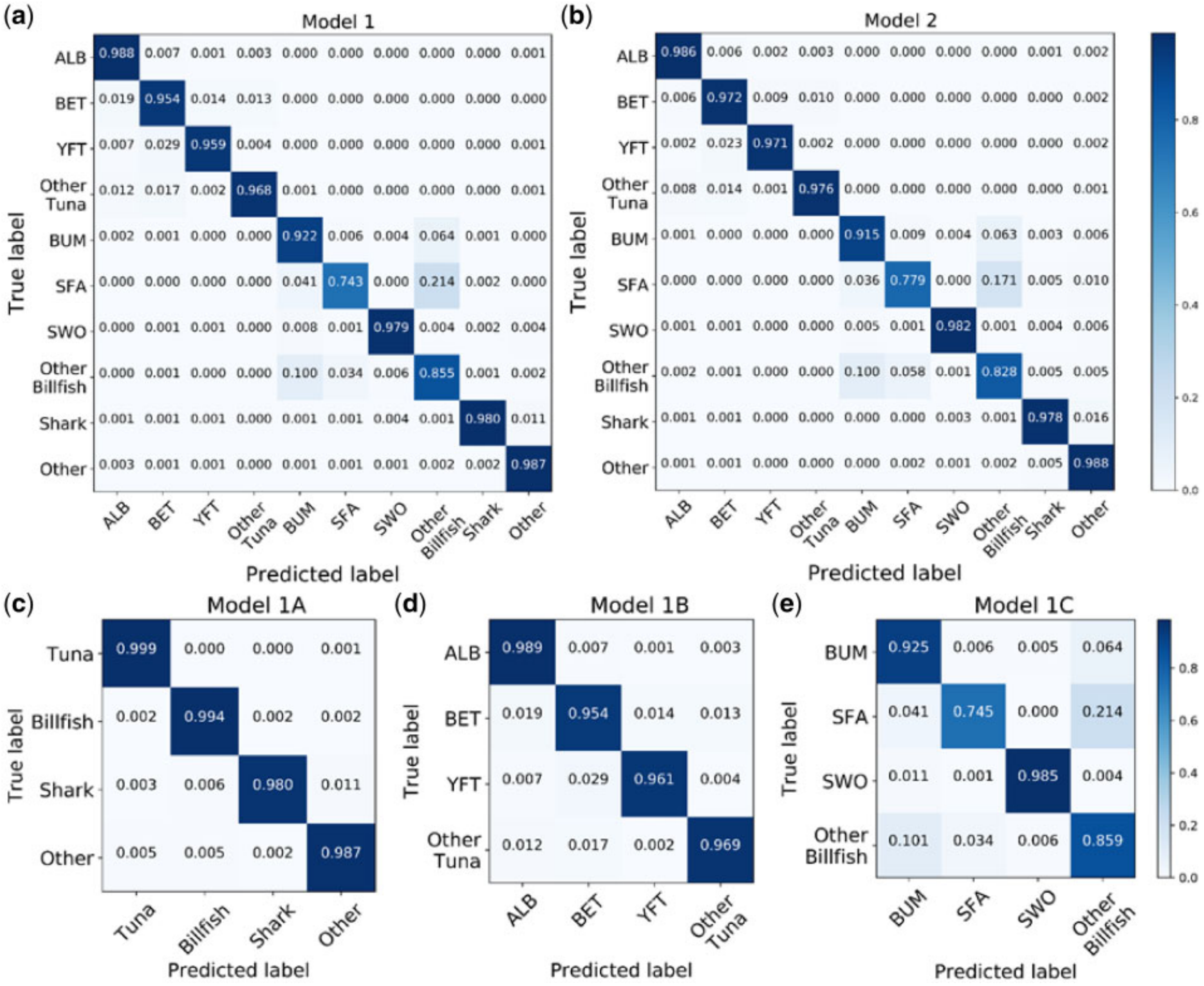
The models were developed using adaptive moment estimation (Kingma and Ba, 2014) as the optimizer and cross-entropy as the loss function. The initial learning rate was set to 0.00002. Each model was trained for 50 epochs. In each epoch, image augmentation was randomly applied to the training images.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

Visualization of filters in the CNN models

Filters of the CNN model were visualized to realize how the CNN models work and what features the models had learned. To visualize a specific filter in a CNN model, a loss function that maximizes the activation of the filter was determined. An image with a dimension of 330 × 250 pixels was next generated and initialized with random pixel values. The gradient of the loss function using the image as the input to the CNN model was calculated. Gradient ascent (Simonyan *et al.*, 2013) was then applied to update the pixel values in the input image. The aforementioned steps were performed for 200 iterations. The resulting input image was the visualization of the filter.

Saliency maps and Grad-CAMs of the CNN models

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



(a) Model 1

True label \ Predicted label	ALB	BET	YFT	Other Tuna	BUM	SFA	SWO	Other Billfish	Shark	Other
ALB	0.988	0.007	0.001	0.003	0.000	0.000	0.000	0.000	0.000	0.001
BET	0.019	0.954	0.014	0.013	0.000	0.000	0.000	0.000	0.000	0.000
YFT	0.007	0.029	0.959	0.004	0.000	0.000	0.000	0.000	0.000	0.001
Other Tuna	0.012	0.017	0.002	0.968	0.001	0.000	0.000	0.000	0.000	0.001
BUM	0.002	0.001	0.000	0.000	0.922	0.006	0.004	0.064	0.001	0.000
SFA	0.000	0.000	0.000	0.000	0.041	0.743	0.000	0.214	0.002	0.000
SWO	0.000	0.001	0.001	0.000	0.008	0.001	0.979	0.004	0.002	0.004
Other Billfish	0.000	0.001	0.000	0.000	0.100	0.034	0.006	0.855	0.001	0.002
Shark	0.001	0.001	0.001	0.000	0.001	0.001	0.004	0.001	0.980	0.011
Other	0.003	0.001	0.001	0.000	0.001	0.001	0.001	0.002	0.002	0.987

(b) Model 2

True label \ Predicted label	ALB	BET	YFT	Other Tuna	BUM	SFA	SWO	Other Billfish	Shark	Other
ALB	0.986	0.006	0.002	0.003	0.000	0.000	0.000	0.000	0.001	0.002
BET	0.006	0.972	0.009	0.010	0.000	0.000	0.000	0.000	0.000	0.002
YFT	0.002	0.023	0.971	0.002	0.000	0.000	0.000	0.000	0.000	0.002
Other Tuna	0.008	0.014	0.001	0.976	0.000	0.000	0.000	0.000	0.000	0.001
BUM	0.001	0.000	0.000	0.000	0.915	0.009	0.004	0.063	0.003	0.006
SFA	0.000	0.000	0.000	0.000	0.036	0.779	0.000	0.171	0.005	0.010
SWO	0.001	0.001	0.000	0.000	0.005	0.001	0.982	0.001	0.004	0.006
Other Billfish	0.002	0.001	0.000	0.000	0.100	0.058	0.001	0.828	0.005	0.005
Shark	0.001	0.001	0.000	0.000	0.000	0.000	0.003	0.001	0.978	0.016
Other	0.001	0.001	0.000	0.000	0.000	0.002	0.001	0.002	0.005	0.988

(c) Model 1A

True label \ Predicted label	Tuna	Billfish	Shark	Other
Tuna	0.999	0.000	0.000	0.001
Billfish	0.002	0.994	0.002	0.002
Shark	0.003	0.006	0.980	0.011
Other	0.005	0.005	0.002	0.987

(d) Model 1B

True label \ Predicted label	ALB	BET	YFT	Other Tuna
ALB	0.989	0.007	0.001	0.003
BET	0.019	0.954	0.014	0.013
YFT	0.007	0.029	0.961	0.004
Other Tuna	0.012	0.017	0.002	0.969

(e) Model 1C

True label \ Predicted label	BUM	SFA	SWO	Other Billfish
BUM	0.925	0.006	0.005	0.064
SFA	0.041	0.745	0.000	0.214
SWO	0.011	0.001	0.985	0.004
Other Billfish	0.101	0.034	0.006	0.859

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

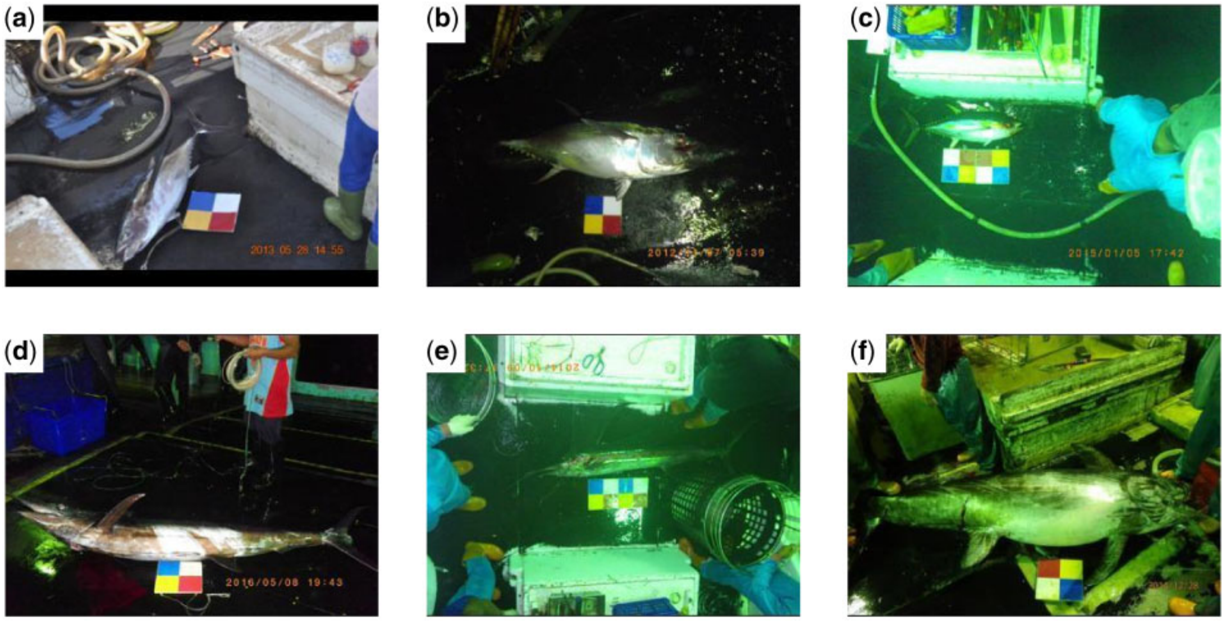


Figure 7. Challenging cases that were successfully identified: (a) ALB, (b) BET, (c) YFT, (d) BUM, (e) SFA, and (f) SWO.

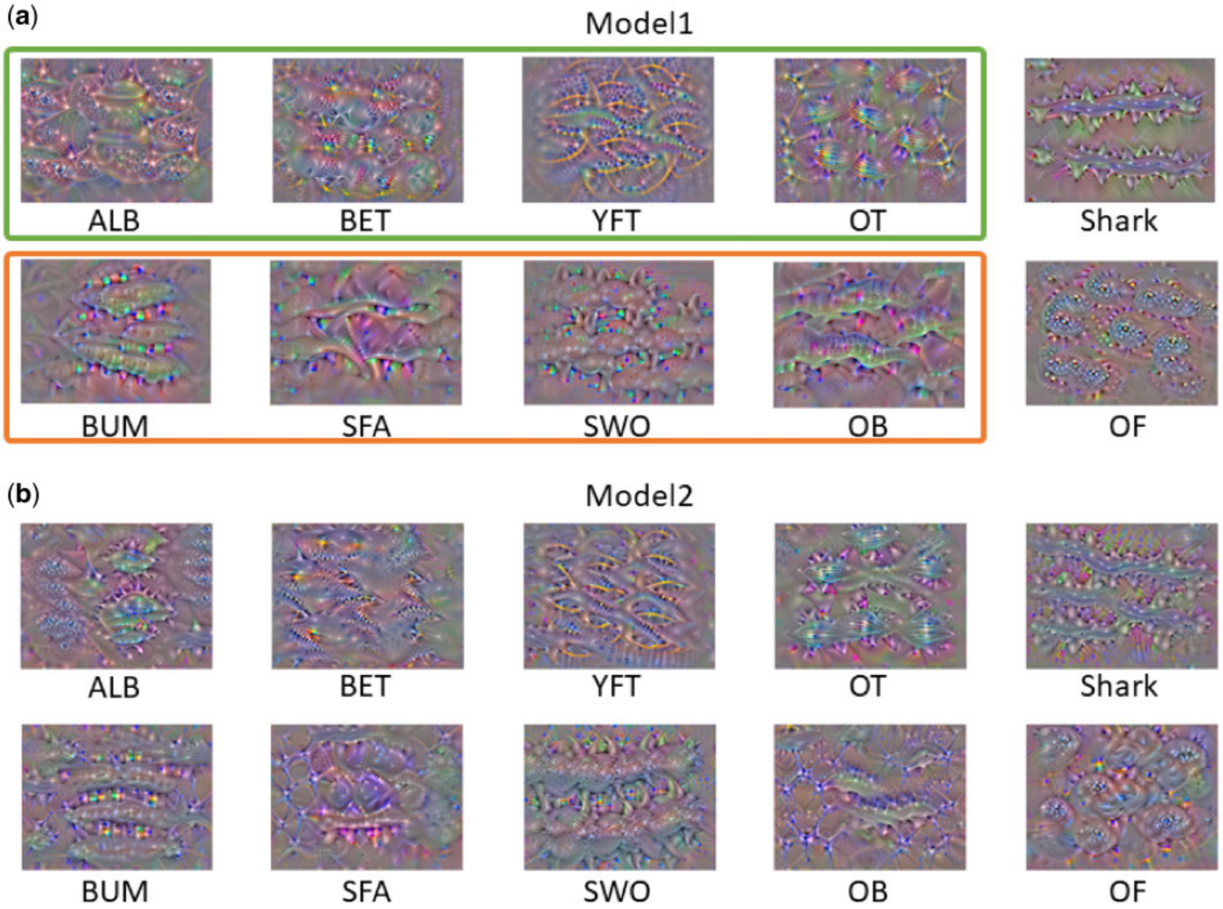


Figure 8. Visualization of the last fully connected filters of each species or type. The green and orange boxes enclose the visualization of filters in Models 1B and 1C, respectively.

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

Fish species identification using bag-of-features approach

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

Results and discussion

Model accuracy and loss during training

The accuracies and losses of the models during training were examined (Figure 5). After 50 epochs, both the training and test losses of Models 1A, 1B, and 2 converged to under 0.16. Both the training and test accuracies of Models 1A, 1B, and 2 reached over 96%. However, for Model 1C, there was ∼6% difference between the training and test accuracies. This observation implied that Model 1C might be slightly overfitted, which could be caused by the inadequate amount of training images (Table 1). The issue of overfitting may be resolved by increasing the amount of the training images of SFA.

Performance of the models

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

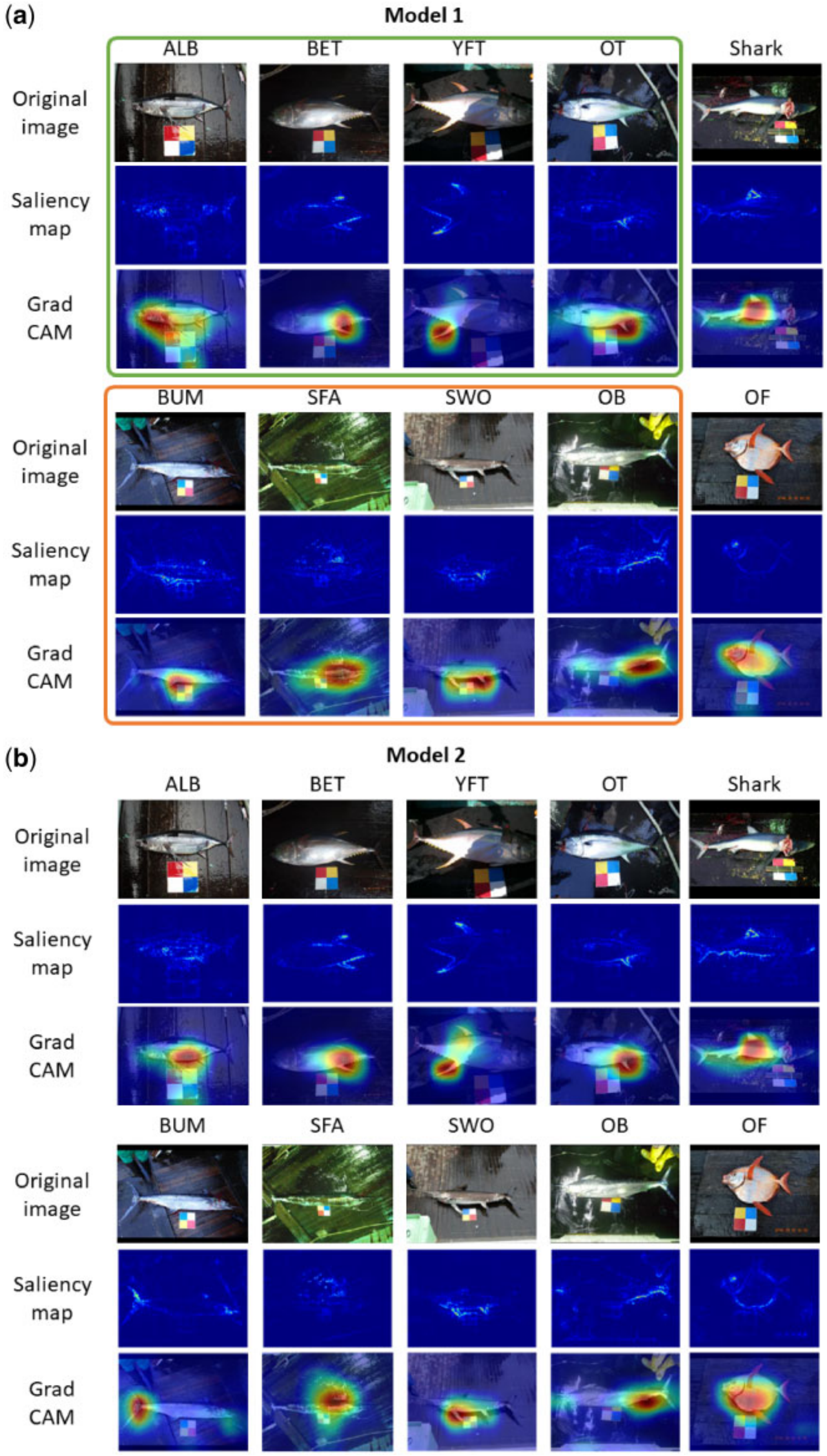


Figure 9. Saliency maps and Grad-CAM of (a) Model 1 and (b) Model 2. The green and orange boxes enclose the visualization of filters in Models 1B and 1C, respectively.

Cases that were challenging to be identified were examined. Figure 7 illustrates the images of ALB, BET, YFT, BUM, SFA, and SWO that were successfully identified. The challenges included panned fish body (Figure 7a), low lamination (Figure 7b and d), colour tone shifting (Figure 7c, e, and f), inadequate resolution (Figure 7c), slanted fish body (Figure 7d), and incomplete fish body (Figure 7f). In Figure 7, the upper jaw of SWO was cut off.

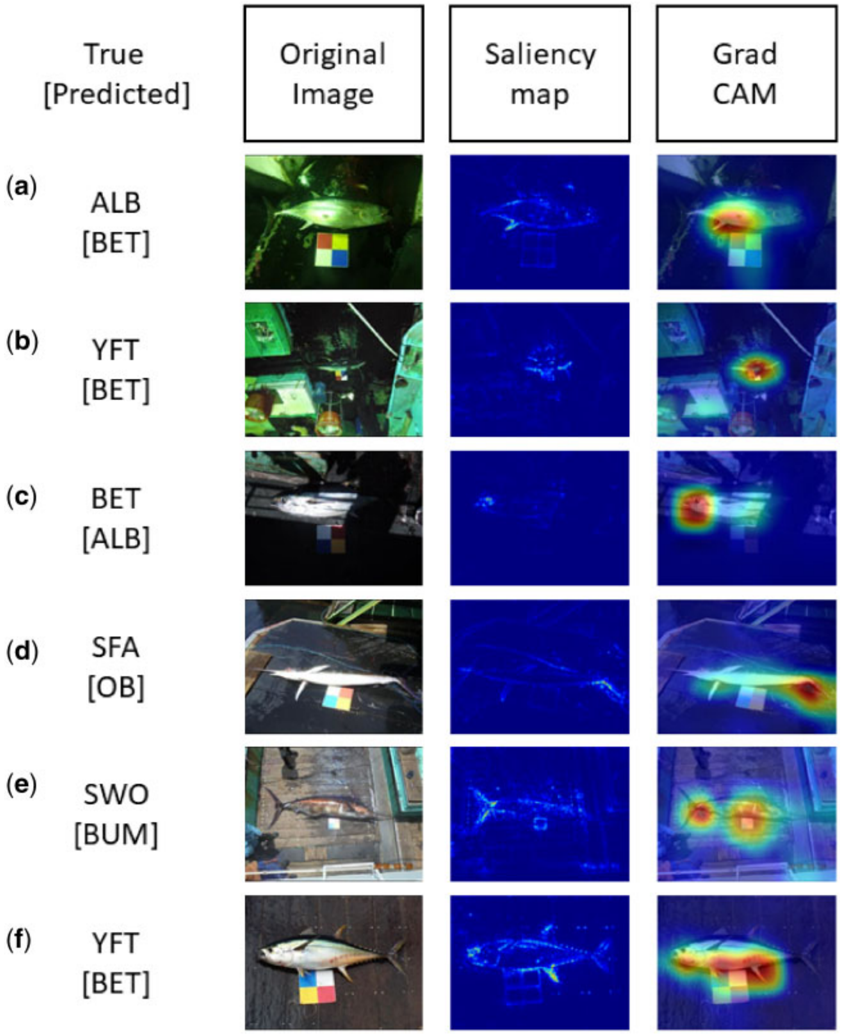


Figure 10. Misclassified cases. The true and predicted categories of the images were shown on the left side.

Filters of the CNN models

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

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

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

Saliency maps and Grad-CAMs of the CNN models

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

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

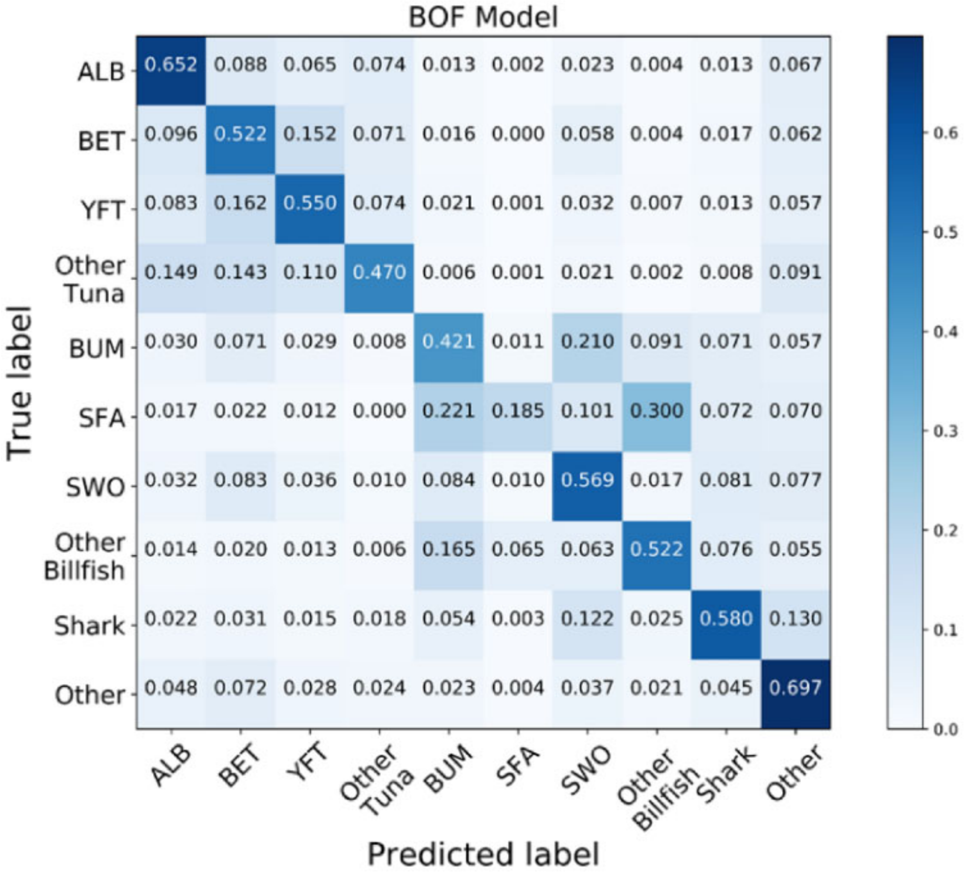


Figure 11. Test accuracy of the BoF model.

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

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

For shark and OF, the first dorsal fins and body contours received considerable attention. The dorsal fins of shark are usually larger than those of tuna, billfish and OF. Moreover, the contours of shark fins are smooth, whereas those of tuna, billfish and OF fins are tippy. This information was used to distinguish shark and OF from tuna or billfish.

Study of misclassification cases

Misclassification occurred due to colour tone variation, inadequate resolution, low illumination, body part occlusion, or fish immaturity. Figure 10a displays an image of ALB that was falsely recognized as BET. The image was acquired at night and was in green tone. The pectoral fin, one of the most essential traits of ALB, of the fish were almost invisible. The saliency map and Grad-CAM of the image confirmed that the pectoral fin received almost no attention. Instead, the anal fin received attention. Figure 10b displays an image of YFT that was falsely recognized as BET. The image was in green tone and was taken from a distance. The saliency map and Grad-CAM of the image indicated that the fish contour was not completely identified. The ventral of the fish received attention at a certain degree. However, the dorsal and anal fins, two of the most essential traits of YFT, received almost no attention. Figure 10c displays an image of BET that was falsely recognized as ALB. Shadow covered the tail of the fish body and made the finlets invisible. The saliency map and Grad-CAM of the image displayed that the fish contour was not completely identified. Although the anterior of the fish received attention at a certain degree, the part typically does not contain traits that are essential for determining the species. Figure 10d displays an image of SFA that was falsely recognized as OB. The body of the fish was tilted so that the dorsal fin, one of the most essential traits of SFA, of the fish was occluded. The saliency map and Grad-CAM of the image displayed that the posterior received attention. However, the posterior of the fish typically does not contain traits that are essential for determining the species. Figure 10e displays an image of SWO that was falsely recognized as BUM. The colour of the second anal fin of the fish was similar to that of the background, making the second anal fin almost invisible. Also, the pectoral fin was close to the fish body, making it almost invisible. The saliency map and Grad-CAM of the image confirmed that the second anal fin or pectoral fin of the fish did not receive strong attention. Figure 10f displays an image of YFT at juvenile stage. The saliency map and Grad-CAM of the image displayed that the contour of the fish was clearly identified and the dorsal and anal fin of the fish received strong attention. However, the lengths of the fins were short. Thus, YFT was falsely recognized as BET. Although misclassified, a tuna species was usually falsely recognized as another tuna species and a billfish species was usually falsely recognized as another billfish species (Figure 10).

The performance of the bag-of-features model

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

Conclusions

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

Funding

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

References

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

Ali-Gombe, A., Elyan, E., and Jayne, C. 2017, August. Fish classification in context of noisy images. *In* International Conference on Engineering Applications of Neural Networks, pp. 216–226. Springer, Cham.

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Lee, S. H., Chang, Y. L., and Chan, C. S. 2017. Lifeclef 2017 plant identification challenge: classifying plants using generic-organ correlation features. Working Notes of CLEF, 2017.

Li, L., and Hong, J. 2014. Identification of fish species based on image processing and statistical analysis research. *In* Mechatronics and Automation (ICMA), 2014 IEEE International Conference on, pp. 1155–1160. IEEE.

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

Lin, M., Chen, Q., and Yan, S. 2013. Network in network. arXiv preprint arXiv:1312.4400v3.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., and Berg, A. C. 2016, October. Ssd: single shot multibox detector. *In* European Conference on Computer Vision, pp. 21–37. Springer, Cham.

Lopez, A. R., Giro-i-Nieto, X., Burdick, J., and Marques, O. 2017, February. Skin lesion classification from dermoscopic images using deep learning techniques. *In* Biomedical Engineering (BioMed), 2017 13th IASTED International Conference on, pp. 49–54. IEEE.

Marini, S., Corgnati, L., Mantovani, C., Bastianini, M., Ottaviani, E., Fanelli, E., Aguzzi, J. *et al.* 2018. Automated estimate of fish abundance through the autonomous imaging device GUARD1. Measurement, 126: 72–75.

Monteagudo, J. P., Legorburu, G., Justel-Rubio, A., and Restrepo, V. 2015. Preliminary study about the suitability of an electronic monitoring system to record scientific and other information from the tropical tuna purse seine fishery. Collective Volumes of Scientific Papers ICCAT, 71: 440–459.

Navarro, A., Lee-Montero, I., Santana, D., Henríquez, P., Ferrer, M. A., Morales, A., Soula, M. *et al.* 2016. IMAFISH_ML: a fully-automated image analysis software for assessing fish morphometric traits on gilthead seabream (*Sparus aurata* L.), meagre (*Argyrosomus regius*) and red porgy (*Pagrus pagrus*). Computers and Electronics in Agriculture, 121: 66–73.

Oquab, M., Bottou, L., Laptev, I., and Sivic, J. 2014. Learning and transferring mid-level image representations using convolutional neural networks. *In* Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1717–1724.

Pan, S. J., and Yang, Q. 2010. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering, 22: 1345–1359.

Rodrigues, M. T., Padua, F. L., Gomes, R. M., and Soares, G. E. 2010, September. Automatic fish species classification based on robust feature extraction techniques and artificial immune systems. *In* Bio-Inspired Computing: Theories and Applications (BIC-TA), 2010 IEEE Fifth International Conference on, pp. 1518–1525. IEEE.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J. 1986. Learning representations by back-propagating errors. Nature, 323: 533.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and Batra, D. 2017. Grad-CAM: visual explanations from deep networks via gradient-based localization. 2017 IEEE International Conference on Computer Vision (ICCV), Venice, 2017, pp. 618–626.

Siddiqui, S. A., Salman, A., Malik, M. I., Shafait, F., Mian, A., Shortis, M. R., and Harvey, E. S. 2017. Automatic fish species classification in underwater videos: exploiting pre-trained deep neural network models to compensate for limited labelled data. ICES Journal of Marine Science, 75: 374–389.

Simonyan, K., Vedaldi, A., and Zisserman, A. 2013. Deep inside convolutional networks: visualising image classification models and saliency maps. arXiv preprint arXiv:1312.6034v2.

Simonyan, K., and Zisserman, A. 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv: 1409.1556v6.

Sivic, J., and Zisserman, A. 2003. Video Google: a text retrieval approach to object matching in videos. *In* null, p. 1470. IEEE.

Sprengel, E., Jaggi, M., Kilcher, Y., and Hofmann, T. 2016. Audio based bird species identification using deep learning techniques. *In* LifeCLEF 2016 (No. EPFL-CONF-229232, pp. 547–559).

Springenberg, J. T., Dosovitskiy, A., Brox, T., and Riedmiller, M. 2014. Striving for simplicity: the all convolutional net. arXiv preprint arXiv:1412.6806v3.

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

Tivive, F. H. C., and Bouzerdoum, A. 2006. Texture classification using convolutional neural networks. *In* TENCON 2006. 2006 IEEE Region 10 Conference, pp. 1–4. IEEE.

Viglione, S. S. 1970. 4 Applications of pattern recognition technology. *In* Mathematics in Science and Engineering, 66, pp. 115–162. Elsevier.

Wei, D., Sahiner, B., Chan, H. P., and Petrick, N. 1995. Detection of masses on mammograms using a convolution neural network. In Acoustics, Speech, and Signal Processing, 1995. ICASSP-95. 1995 International Conference on, 5, pp. 3483–3486. IEEE.

Yosinski, J., Clune, J., Bengio, Y., and Lipson, H. 2014. How transferable are features in deep neural networks?. *In* Advances in Neural Information Processing Systems, pp. 3320–3328.

Handling editor: Cigdem Beyan