CONVOLUTIONAL NEURAL NETWORK ARCHITECTURES PERFORMANCE EVALUATION FOR FISH SPECIES CLASSIFICATION

HAMIZAH ISMAIL[1]*, AHMAD FAISAL MOHAMAD AYOB[1], AIDY @ MUHAMED SHAWAL M MUSLIM[2] AND MOHAMAD FAKHRATUL RIDWAN ZULKIFLI[1]

[1]*Faculty of Ocean Engineering and Informatics, [2]Institute of Oceanography and Environment, Universiti Malaysia Terengganu, Kuala Nerus, 21030 Terengganu, Malaysia.*

Corresponding author: hamizahismail929@gmail.com

Abstract: Fish image classification tool is important in the field of ichthyology. In this paper, we present a fish image classification benchmark comparison across different types of convolutional neural network (CNN). CNN extracts features from labeled image data to solve classification problems. CNN models were trained to classify fish images using transfer learning with data augmentation. CNN models consisting of AlexNet, GoogLeNet, and ResNet were incorporated in the benchmark tests. A dataset of 18,000 fish images across 18 categores, were split into 5,400 images for validation (30%) and 12,600 images for training (70%) dataset. Such neural network models show high accuracy up to 99.85% (AlexNet), 96.39% (GoogLeNet) and 99.51% (ResNet-50). To evaluate the performance of each framework, the analysis presented consists of classification accuracy, learning curve, validation test, top-five prediction and confusion matrix. The work presented here has shown its potential to contribute towards accurate development of state-of-the-art fish classification tools. It is envisioned that these CNN algorithms have the potential to assist in fish image classification problems with high accuracy despite visually similar features of images in the dataset.

Keywords: Fish image classification, marine technology, artificial intelligence, deep learning.

Introduction

The development of automated fish classification tools is the most recent trend in machine learning (Man, 2016 & Sarigül *et al.,* 2017). Fish classification tasks are beneficial to aid individuals without taxonomic backgrounds, such as data collectors and entry-level ichthyologists. Fish image classifications were explored using machine learning algorithms. Serna and Segura (2014), designed an algorithm and defined a range of new features that use pattern recognition with artificial neural networks (ANN) for Europe and South America fish species identifications with 91.65% accuracy. Deep artificial neural networks with a different number of layers and various filter sizes are used for comparison of different deep structures for fish classification tasks in Queensland University of Technology fish dataset (Sarigül & Avci, 2017).

Recently, Convolutional Neural Networks (CNNs) have shown a remarkable accuracy for image classification (Krizhevsky *et al.,* 2012; Russakovsky *et al.,* 2015). Deep learning extracts features from the image dataset with different layered architectures consisting of convolutional layers, pooling layers, and finally, connected to a fully connected layer to produce the output. Currently, CNN has been studied by many researchers and applied to a wide range of classification problems. There are some works that applied CNNs in numerous studies, as shown in Table 1. In fish classification, the use of CNN is challenging due to the visually similar features of fish. The CNN is proposed to provide high accuracy in fish classification as it learns the boundary and color features of the fish images. Table 2 shows the fish image classification approach using CNN.

Table 1: Summary of related deep learning works for other types of image classification

References	Categories	Data	Architecture	Results
(Liu, 2020)	Soft-shell shrimp and sound shrimp	Images taken using CCD camera	Deep-ShrimpNets-12	Mean accuracy precision of 97.20%
(Thenmozhi *et al.,* 2019)	40, 24 and 40 insect classes	NBAIR, Xie1 and Xie2 insect dataset	CNN with 6 convolution layers and 5 pooling layers	Accuracy of 96.75% (NBAIR), 97.47% (Xie1) and 95.97% (Xie2)
(Steinbrener *et al.,* 2019)	13 classes of fruit and vegetable	Images taken using a hyperspectral camera	Convolutional layer with 3 kernels of size (3x3x5) and rectified linear unit as activation	Accuracy of 92.23%
(Ríos *et al.,* 2019)	14 classes of coral species	RSMAS small coral texture dataset	ResNet-152 with data augmentation (zooming)	Accuracy of 98.63%
(Kozłowski *et al.,* 2019)	6 varieties	Scanned images of barley	Convolutional layers with 64 filters of size 3x3	Accuracy of 93.21%
(Xu & Sun, 2018)	Normal flesh and gaping categories	Salmon muscle gaping images taken from digital camera	AlexNet	Correct classification rates (CCRs) of 91.60% for test data set
(Wang *et al.,* 2018)	6 types of plants	Original images of crop and weed captured by UAV	CNN with 5 layers	Accuracy 95.60% for 300x300 input image resolutions
(Tyagi *et al.,* 2018)	11 classes of phenotypical deformation	Phenotypes zebrafish embryo obtained using high-resolution imaging	Fine-Tuned VGG16	Accuracy 83.55%
(Verma & Gupta, 2018)	20 species of wild animal	Images taken using camera trap	Deep CNN	Accuracy of 91.40%
(Dangelo & Tennefoss, 2015)	121 classes of plankton	Gray-scale images of photo plankton species	Network in Network	26.10% correct class prediction

For pre-trained CNN models in this study, AlexNet, GoogLeNet, and Residual Net (ResNet) were utilized. The AlexNet architecture stacked multiple convolutional layers that allow for more complex features of the input vector to be selected (Krizhevsky *et al*., 2012). It contains eight learned layers with weights made up of five convolutional layers, max-pooling layers, dropout layers, and three fully connected layers. AlexNet used a novel activation function, Rectified Linear Unit (ReLU) that reduces its training time. ReLU is applied to the output of every convolutional layer and fully connected layer. Local Response Normalization (LRN) implements competition between neurons at the same location but in different feature maps. LRN follows the first and second convolutional layers (Krähenbühl, *et al*., 2015). Data augmentation and dropout layer were also implemented in the network to prevent overfitting of data.

Table 2: Summary of related works using deep learning for fish image classification

References	Categories	Data	Architecture	Results
(Liawatimena *et al.,* 2019)	3 species	Fish images	AlexNet	Accuracy of 99.63%
(Kratzert *et al.,* 2018)	15 species	Images captured from videos	VGG-16	Accuracy of 93.00%
(Rathi, Jain, & Indu, 2018)	23 species	Images captured from videos	CNN	Accuracy of 96.29%
(Siddiqui *et al.,* 2018)	16 species	Images captured from videos	CNN	Accuracy of 94.30%
(Polzounov *et al.,* 2016)	1 species	Right whale images	CNN	Accuracy of 99.18%
(Salman *et al.,* 2016)	15 species	Images captured from videos	CNN	Accuracy of 90.00%
(Choi, 2015)	15 species	Images captured from videos	GoogLeNet	Precision of 81.00%

Moreover, the GoogLeNet has 22 layers of deep network with 9 inception units and a fully connected layer before the output. Inception units or modules are inspired by inception layers that have variable receptive fields that are created by different kernel sizes (Alom *et al*., 2018). The architecture uses a combination of inception modules, each including pooling, convolutions at different scales and filter concatenation operations. The inception module has two layers and 6 convolutional blocks that are an essential component of GoogLeNet. It uses the parallel combination of 1x1, 3x3 and 5x5 filters; in GoogLeNet, the 1x1 filter works as a feature selector that reduces the number of parameters, which reduces the processing speed (Szegedy *et al*., 2015).

In addition, the ResNet introduces a principled approach to add shortcut connections for every two layers to a VGG style-network. ResNet is developed with many different numbers of layers such as 34, 50, 101, and 152. ResNet-50 is the most popular network consisting of 50 layer residual networks that have 50 layers with 49 convolutional layers and a final fully connected classification layer. The ResNet network consists of several basic residual blocks; ReLU activation functions and convolution layers. The pre-trained CNN and its properties are presented in Table 3.

In this study, we used the CNN architectures with different depths and structures to determine performance evaluation for fish image classification using transfer learning. Transfer learning was evident to be a remarkable and effective way to train a large network using insufficient training data without overfitting (Huang, Pan, & Lei, 2017). The different CNN architectures are investigated until the latest well-known architectures published in 2017; AlexNet, GoogLeNet, and ResNet-50. The transfer learning properties of Convolutional

Table 3: Pre-trained CNN models with its properties

References	Architecture Layers	Parameters (Millions)	Image Input Size (pixels)
AlexNet (Krizhevsky *et al*., 2012)	8	61.0	227x227
GoogLeNet (Szegedy *et al*., 2015)	22	7.0	224x224
ResNet-50 (Wu *et al*., 2017)	50	25.6	224x224

Neural Networks are exploited for fish image classification. Then, three different CNN architectures (AlexNet, GoogLeNet, and ResNet-50) are trained and evaluated on fish species images dataset, namely FishNet. The performances of pre-trained CNN models for fish species classification are compared specifically, using a new test image.

Materials and Methods

Data Collection
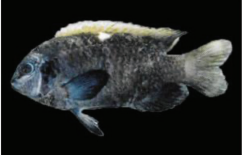
FishNet dataset was collected through Google search engine and QUT fish dataset (Anantharajah *et al*., 2014). It consists of real-world images of fish captured in conditions defined as controlled, out-of-the-water and in-situ. In our experiments, a total of 736 images were pre-processed via data augmentation method, which resulted in 29,275 fish images (Table 4). For each fish species, there are between 1,000 and 2,800 of images. However, in network training, the lowest number, which is 1000 images in each category, is automatically selected to avoid bias. Thus, a total of 18,000 fish images were used for this experiment. Such a number of images were divided randomly (70%-30%) into training images (12,600) images and test images (5,400) set, respectively. The original size of each image is not more than 600x600 pixels, with random dimensions of images. Then, the collection of data images was down-sampled to the same dimensions as the images used for the pre-trained network, i.e. 227x227 pixels for AlexNet and 224x224 pixels for GoogLeNet, and ResNet-50 respectively.

Table 4: Fish sample images dataset of different species.

Species	Sample Images	Total Images
Atule mate		1760
Caranx ruber		1950
Cetoscarus bicolor		1640
Decapterus russelli		1000
Epinephelus areolatus		1404
Epinephelus fasciatus		2800

Halichoeres argus		1800
Hemiglyphidodon plagiometopon		1000
Lutjanus inermis		1120
Megalaspis cordyla		1000
Monodactylus argenteus		2480
Parastromateus niger		2040
Pomacentrus alexanderae		1560
Rastrelliger kanagurta		2161
Salmo salar		1560

Scomber scombus		1600
Scomberoides commersonnianus		1080
Scomberoides tala		1320

Data Pre-processing

Data augmentation is an image pre-processing step before serving as data input to the network. This step is very important to prevent the overfitting of the network, hence increasing the robustness of the CNN models. As explained in the previous section, 736 images were augmented to enrich the dataset size in terms of volume and orientation. The training images were augmented by 36 rotations; in 10º rotation increments. Figure 1 shows an example of data augmentation, where; (a) is the original image, (b)-(j) are the rotated images. After the augmentation, 29,275 images were generated.

(a) Original image (b) 10-degree rotation (c) 20-degree rotation (d) 30-degree rotation (e) 40-degree rotation

(f) 50-degree rotation (g) 60-degree rotation (h) 70-degree rotation (i) 80-degree rotation (j) 90-degree rotation

Figure 1: Data augmentation examples

Case Study

Transfer learning is an approach that takes layers from a network trained on a large dataset, transfer weights and layers from pre-train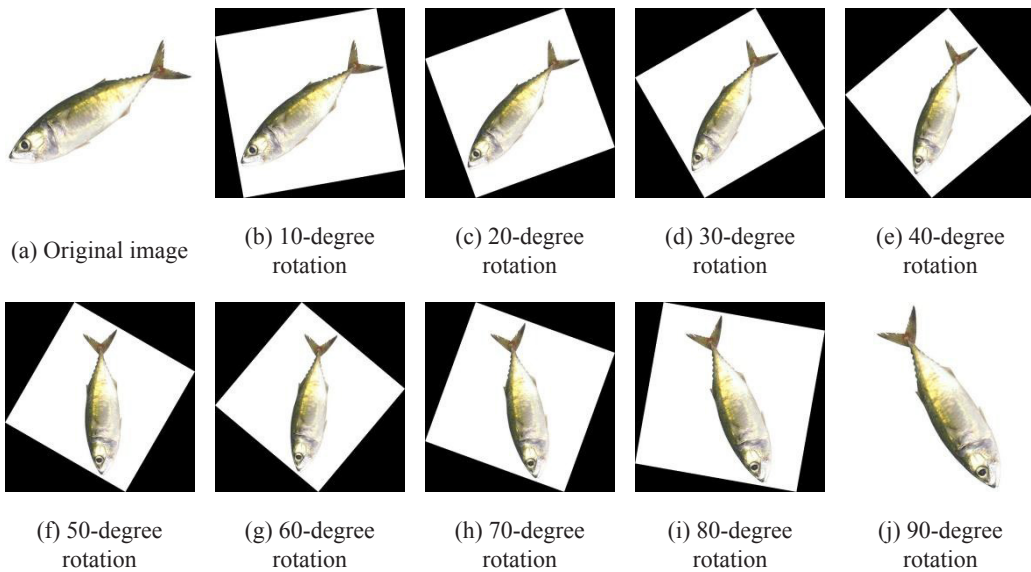ed models then replace the final classification layer and fine-tune on the image classification task. Initial pre-trained networks are trained on a subset of ImageNet database, on more than a million images to classify images into 1,000 categories. In this case study, we compare the performance evaluations of different CNNs using transfer learning method to classify images into 18 fish species. Illustrated in Figure 2 is the framework of a pre-trained CNN case study for fish image classification using transfer learning.

The dataset of labeled fish species images is resized with respect to the pre-trained image size requirement following the pre-trained model properties in Table 3. The input image size is denoted as the spatial dimensional, (width and height) times the number of 3 of colour components; red, green and blue. Next, the pre-trained CNNs are chosen. Three final layers of the pre-trained network (fully connected layer, softmax layer, and classification layer) are replaced with three new layers to adapt to the new dataset. Since the initial network learned a rich set of image features, fine-tuning a network with transfer learning is able to learn features specific to the new dataset. After that, train the pre-trained CNNs on the FishNet dataset to predict and assess the network accuracy.

The experiments were run on a desktop computer with Intel-i7 processor and 64 GB of RAM. During training, the hyper-parameters are adjusted to minimize training loss, which constitutes classification error. The initial learning rate was 0.0001. A similar batch size, 100, is employed and trained on 20 epochs to compare training time for each network. The network was trained with stochastic gradient descent with 0.9 momentums and the training was terminated after 4,080 iterations. CNN final output comprised a dimension, where n represents the number of fish species. Through

the fully connected layer, the network is able to transform the original image input to produce class scores for classification accuracy performance. Finally, a new test image is fed into the network to compare performance evaluations.

In this paper, we discuss fish image classification methods that utilize the softmax function. Softmax is a classifier that is able to produce class-probabilities (Springenberg *et al.,* 2014). The softmax layer gives the output as the probability distribution over different labels. The softmax function is often used in multiple classification logistic regression models. The softmax function specifies a discrete problem probability distribution for *K* classes, defined as (Agarap, 2018):

$$\sum_{K=1}^{K} \rho k \qquad (1)$$

The equation denotes as the activation at the penultimate layer of a neural network, and as its weight parameters at the softmax layer, thus producing **o** as the input to the softmax layer.

$$o = \sum_{i}^{n-1} \theta_i x_i \qquad (2)$$

Consequently, the equation will be:

$$\rho k = \frac{exp\ (o_k)}{\sum_{k=0}^{n-1}\ exp\ (o\_k)} \qquad (3)$$

Hence, the predicted class would be $\hat{y}$,

$$\hat{y} = \frac{arg\ arg\ max}{i \in 1,...,N} \rho i \qquad (4)$$

Confusion matrix is a table to describe classification accuracy performance. A confusion matrix is commonly known as a contingency table in which the matrix could be arbitrarily large (Santra & Christy, 2012). Basic model performance measures, such as accuracy, error-rate, specificity, sensitivity, and precision, are derived from the confusion matrix. The basic measure classifiers that are most widely used are accuracy (ACC) and error rate (ERR) (Saito & Rehmsmeier, 2015). The accuracy is defined as;

$$Accuracy = \frac{(True\ Positive\ +\ True\ Negative)}{(True\ Positive +\ True\ Negative\ +\ False\ Negative\ +\ False\ Positive)} \qquad (5)$$
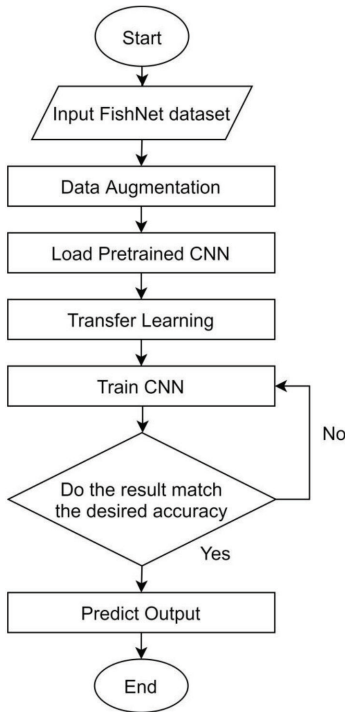
Figure 2: Flowchart of fish image classification study

Result and Discussion

This experiment was aimed to evaluate the performance of different pre-trained CNN using transfer learning when predicting the fish species. The performance of the models for fish image classification through overall classification accuracy, learning curves, classification accuracy from the validation dataset, top-five accuracy from test dataset and confusion matrix was analyzed. Table 5 shows the accuracy and training time for all pre-trained networks used in this study. It is observed that the best performance was obtained by AlexNet, followed by ResNet-50 and GoogLeNet, with

an accuracy of 99.85%, 99.51%, and 96.39% respectively. The classification accuracy is the percentage of correctly recognized fish species. Regarding the different pre-trained CNN, better performance was obtained with the AlexNet model, followed by ResNet-50 and GoogLeNet. This result shows that the performance of methods is related to the complexity of the network. The additional layers or more complex architecture will not ensure higher accuracy of prediction (Ayob *et al.,* 2019).

Table 5 shows the training time of each pre-trained network obtained by AlexNet, GoogLeNet, and ResNet-50, with duration of 352 minutes 49 seconds, 662 minutes 59 seconds, and 1,743 minutes 27 seconds respectively. AlexNet requires the least training time to complete the learning methods due to the lowest depth of the network. It can be seen that the training time was tied to the total number of architecture layers of each pre-trained CNN; AlexNet had 8 layers, GoogLeNet had 22 layers and ResNet-50 had 50 layers. Fewer layer in the network contributes to better fish image classification performance.

The overall performance of models was analyzed by using a learning curve. The learning curve provides mathematical representations of the training plot that takes place as task classification. Figure 3 illustrates the learning curves for monitoring the performance of fish image classification tasks. Figure 3(a), Figure 3(b) and Figure 3(c) show the learning curve of training loss and validation loss over a number of iterations for AlexNet, GoogLeNet and ResNet-50 respectively. Figure 3(a) illustrates the plot of training loss decreasing to a point of stability. Meanwhile the plot of validation loss decreases to a point of stability and has

Table 5: Performance comparison of each pre-trained networks

Pre-trained Networks	Classification accuracy (%)	Training Time
AlexNet	99.85	352 min 49 sec
GoogLeNet	96.39	662 min 59 sec
ResNet-50	99.51	1743 min 27 sec

a minimal gap with the training loss. Figure 3(b) shows both the plot of training loss and validation loss decreasing to a point of stability and has a small gap between the two lines. In Figure 3(c), below, it can be seen that the plot of training loss and validation loss remain stable and has the smallest gap among the pre-trained CNN. Regarding the learning curves, it can be seen that the performance of AlexNet, GoogLeNet and ResNet-50 are remarkably fitted, with sufficient information of training and validation dataset to learn the fish image classification task.
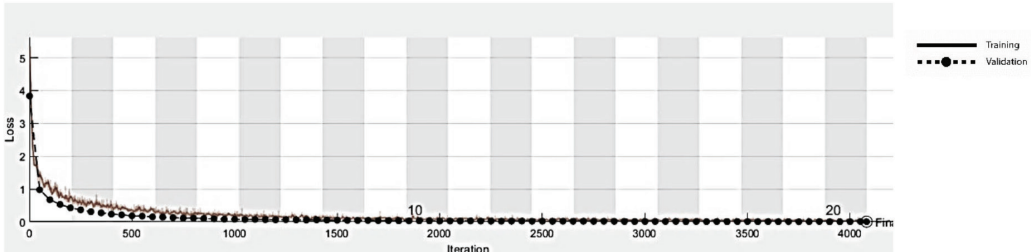
Figure 3(a): The training plot of loss over the number of iterations for AlexNet
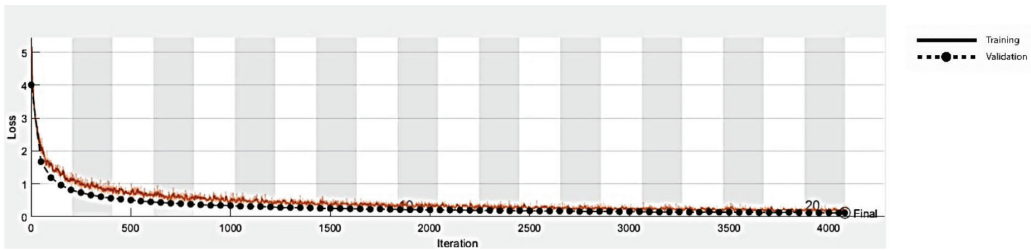
Figure 3(b): The training plot of loss over the number of iterations for GoogLeNet
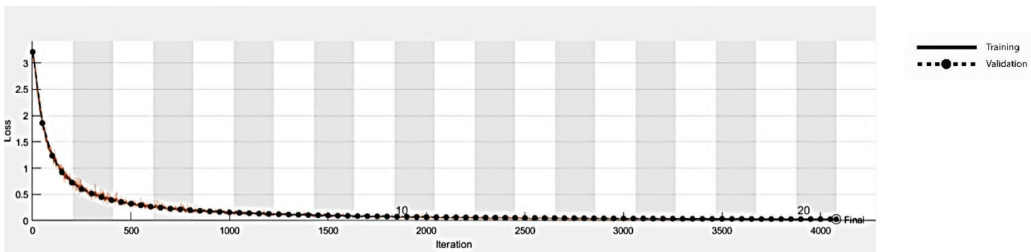
Figure 3(c): The training plot of loss over the number of iterations for ResNet-50

After that, deep learning models for each type of fish were evaluated using a validation image from each species. A validation set is used to monitor the model performance during the training process and fine tune hyper parameters. The accuracy results for each type of fish species are tabulated in Table 6. It can be seen that AlexNet has the best accuracy as it shows better classification in comparison to the other architecture, whereas the least accurate architecture is ResNet-50. The results show that AlexNet has 100% correctly classified 11 fish species out of 18 species. Meanwhile, ResNet-50 has incorrectly classified 2 fish species that are *Caranx ruber* and *Scomber scombrus* which was misclassified as *Rastrelliger kanagurta*. This is the expected result, given the visually similar features of fish images in the dataset. Both *Epinephelus fasciatus* and *Monodactylus argenteus* have 100% accuracy on all the pre-trained CNN as it has a significant color feature that differentiates them from other species.

Table 6: Accuracy results for each type of fish on a validation dataset

	AlexNet (%)	GoogLeNet (%)	ResNet-50 (%)
Atule mate	91.70	73.00	62.20
Caranx ruber	83.60	74.00	(91.70)
Cetoscarus bicolor	100.00	99.80	100.00
Decapterus russelli	97.40	91.00	93.80
Epinephelus areolatus	100.00	99.90	99.30
Epinephelus fasciatus	100.00	100.00	100.00
Halichoeres argus	99.90	88.70	98.20
Hemiglyphidodon plagiometopon	100.00	98.60	92.80
Lutjanus inermis	100.00	98.90	96.00
Megalaspis cordyla	99.70	53.90	99.90
Monodactylus argenteus	100.00	100.00	100.00
Parastromateus niger	100.00	99.20	99.90
Pomacentrus alexanderae	100.00	99.30	95.10
Rastrelliger kanagurta	99.90	93.00	99.90
Salmo salar	100.00	93.00	99.90
Scomber Scombrus	100.00	93.80	(63.60)
Scomberoides commersonnianus	63.30	50.50	66.10
Scomberoides tala	100.00	98.70	95.90

After that, performance of the pre-trained model with an image of test data (Figure 4) was evaluated. Figure 5 displays the top five predicted labels and their associated probabilities. The results suggest that ALexNet correctly classifies the image as *Rastrelliger kanagurta* with 69.9% probability and

Figure 4: A test image of *Rastrelliger kanagurta*

ResNet-50 classifies the image as a *Rastrelliger kanagurta* with a 30.0% probability, which is misclassified as *Atule mate*. GoogLeNet classifies the image as a *Rastrelliger kanagurta* with the lowest probability (3.5%). It can be seen that AlexNet model shows a high probability of correctly classifying fish species, while GoogLeNet and ResNet-50 were incorrect. This is expected as the test image was visually similar to the characteristic of *Atule mate* and has almost the same color and shape features.

The confusion matrix obtained from this experiment is shown in Figure 6, where along the x-axis are listed the target class labels and along the y-axis are the output class predictions. Along the first diagonal section are the correct classifications, meanwhile, all the other entries show classification incorrectly. The bottom right cell of the confusion matrix shows the overall classification accuracy. In this confusion matrix, Figure 6(a), it can be
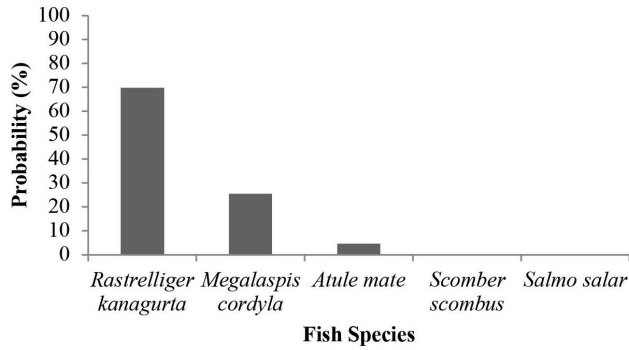
Figure 5(a): AlexNet correctly classified the image as a *Rastrelliger kanagurta* with 69.9% probability
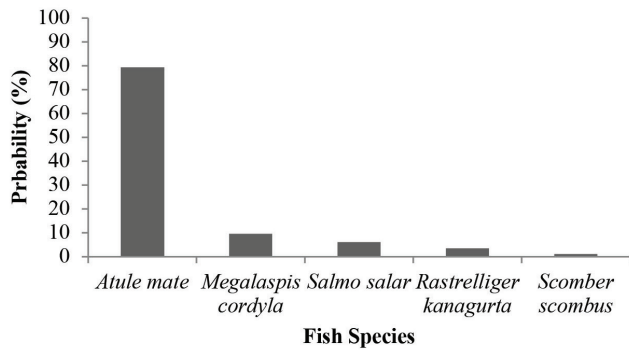
Top 5 Prediction (GoogLeNet)

Figure 5(b): GoogLeNet classified the image as a *Rastrelliger kanagurta* with a 3.5% probability and classified incorrectly as *Atule mate*
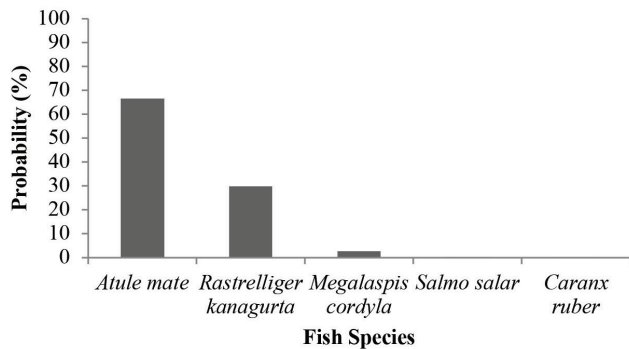
Top 5 Prediction (ResNet-50)

Figure 5(c): ResNet-50 classified the image as a *Rastrelliger kanagurta* with a 30.0% probability and classified incorrectly as *Atule mate*

seen that *Decapterus russelli* and *Megalaspis cordyla* have 3 examples wrongly predicted as *Rastrelliger kanagurta*. Figure 6(b) shows that *Atule mate* has 35 examples wrongly predicted as *Rastrelliger kanagurta*. Additionally, from the classification, *Decapterus russelli* has 29 examples wrongly predicted as *Rastrelliger kanagurta*. As illustrated in Figure 6(c), *Atule mate* has 6 examples wrongly predicted as *Rastrelliger kanagurta*. It can be observed that both GoogLeNet and ResNet-50 have incorrectly classified *Atule mate* as *Rastrelliger kanagurta* as shown in the top-five accuracy result.

Confusion Matrix

Output Class (rows) × Target Class (columns). Each cell shows count and percentage.

Output Class \ Target	Atule mate	Caranx ruber	Cetoscarus bicolor	Decapterus russelli	Epinephelus areolatus	Epinephelus fasciatus	Halichoeres argus	Hemiglyphidodon plagiometopon	Lutjanus inermis	Megalaspis cordyla	Monodactylus argenteus	Parastromateus niger	Pomacentrus alexanderae	Rastrelliger kanagurta	Salmo salar	Scomber scombus	Scomberoides commersonnianus	Scomberoides tala	(acc)
Atule mate	528 / 6.0%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100% / 0.0%
Caranx ruber	0	583 / 6.6%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	99.8% / 0.2%
Cetoscarus bicolor	0	0	492 / 5.6%	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100% / 0.0%
Decapterus russelli	0	0	0	297 / 3.4%	0	0	0	0	0	0	0	0	0	0	0	0	0	1	99.7% / 0.3%
Epinephelus areolatus	0	0	0	0	420 / 4.8%	0	0	0	0	0	0	0	0	0	0	0	0	0	100% / 0.0%
Epinephelus fasciatus	0	0	0	0	0	840 / 9.6%	0	0	0	0	0	0	0	0	0	0	0	0	100% / 0.0%
Halichoeres argus	0	0	0	0	0	0	540 / 6.1%	0	0	0	0	0	0	0	0	0	0	0	100% / 0.0%
Hemiglyphidodon plagiometopon	0	0	0	0	0	0	0	300 / 3.4%	0	0	0	0	0	0	0	0	0	0	100% / 0.0%
Lutjanus inermis	0	2	0	0	1	0	0	0	336 / 3.8%	0	0	0	0	0	0	0	0	0	99.1% / 0.9%
Megalaspis cordyla	0	0	0	0	0	0	0	0	0	297 / 3.4%	0	0	0	0	0	0	0	0	100% / 0.0%
Monodactylus argenteus	0	0	0	0	0	0	0	0	0	0	744 / 8.5%	0	0	0	0	0	0	0	100% / 0.0%
Parastromateus niger	0	0	0	0	0	0	0	0	0	0	0	612 / 7.0%	0	0	0	0	0	0	100% / 0.0%
Pomacentrus alexanderae	0	0	0	0	0	0	0	0	0	0	0	0	468 / 5.3%	0	0	0	0	0	100% / 0.0%
Rastrelliger kanagurta	0	0	0	3	0	0	0	0	0	3	0	0	0	648 / 7.4%	0	0	0	1	98.9% / 1.1%
Salmo salar	0	0	0	0	0	0	0	0	0	0	0	0	0	0	468 / 5.3%	0	0	0	100% / 0.0%
Scomber scombus	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	480 / 5.5%	0	1	99.8% / 0.2%
Scomberoides commersonnianus	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	324 / 3.7%	0	100% / 0.0%
Scomberoides tala	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	392 / 4.5%	100% / 0.0%
(acc)	100% / 0.0%	99.7% / 0.3%	100% / 0.0%	99.0% / 1.0%	99.8% / 0.2%	100% / 0.0%	100% / 0.0%	100% / 0.0%	99.0% / 1.0%	100% / 0.0%	100% / 0.0%	100% / 0.0%	100% / 0.0%	100% / 0.0%	100% / 0.0%	100% / 0.0%	99.0% / 1.0%	99.9% / 0.1%	

Figure 6(a): Confusion matrix for all classes and all attributes (AlexNet)

Confusion Matrix

Output Class \ Target Class	Atule mate	Caranx ruber	Cetoscarus bicolor	Decapterus russelli	Epinephelus areolatus	Epinephelus fasciatus	Halichoeres argus	Hemiglyphidodon plagiometopon	Lutjanus inermis	Megalaspis cordyla	Monodactylus argenteus	Parastromateus niger	Pomacentrus alexanderae	Rastrelliger kanagurta	Salmo salar	Scomber scombus	Scomberoides commersonnianus	Scomberoides tala	Accuracy
Atule mate	463	5	0	1	0	0	0	0	0	2	0	0	0	6	1	4	6	4	94.1% / 5.9%
Caranx ruber	10	564	3	0	1	0	0	0	0	8	0	1	5	0	1	0	1	2	94.6% / 5.4%
Cetoscarus bicolor	1	0	489	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	99.6% / 0.4%
Decapterus russelli	1	0	0	250	0	0	0	0	0	6	0	0	0	0	1	1	5	3	93.6% / 6.4%
Epinephelus areolatus	0	0	0	0	419	0	1	2	1	0	0	0	0	0	1	0	0	0	98.8% / 1.2%
Epinephelus fasciatus	1	0	0	0	0	840	0	0	1	0	0	0	0	0	1	0	0	0	99.6% / 0.4%
Halichoeres argus	0	0	0	0	0	0	538	0	1	0	0	0	0	0	0	0	0	0	99.8% / 0.2%
Hemiglyphidodon plagiometopon	0	0	0	0	0	0	1	296	0	0	0	0	4	0	0	0	0	0	98.3% / 1.7%
Lutjanus inermis	1	1	0	0	0	0	0	0	333	0	0	2	0	0	2	0	0	0	98.2% / 1.8%
Megalaspis cordyla	1	4	0	3	0	0	0	0	0	253	0	0	0	2	0	0	10	1	92.3% / 7.7%
Monodactylus argenteus	0	0	0	0	0	0	0	0	0	0	743	2	2	1	0	0	0	0	99.3% / 0.7%
Parastromateus niger	0	0	0	0	0	0	0	0	0	1	1	606	0	0	0	0	0	0	99.7% / 0.3%
Pomacentrus alexanderae	0	0	0	0	0	0	0	2	0	0	0	0	457	0	0	0	0	0	99.6% / 0.4%
Rastrelliger kanagurta	35	10	0	29	0	0	0	0	0	16	0	1	0	636	1	3	4	15	84.8% / 15.2%
Salmo salar	0	0	0	4	0	0	0	0	0	4	0	0	0	0	459	6	2	0	96.6% / 3.4%
Scomber scombus	2	0	0	3	0	0	0	0	0	9	0	0	0	0	1	466	1	1	96.5% / 3.5%
Scomberoides commersonnianus	2	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	289	6	96.7% / 3.3%
Scomberoides tala	11	1	0	9	0	0	0	0	0	0	0	0	0	3	0	0	6	364	92.4% / 7.6%
Accuracy	87.7% / 12.3%	96.4% / 3.6%	99.4% / 0.6%	83.3% / 16.7%	99.5% / 0.5%	100% / 0.0%	99.6% / 0.4%	98.7% / 1.3%	99.1% / 0.9%	84.3% / 15.7%	99.9% / 0.1%	99.0% / 1.0%	97.6% / 2.4%	98.1% / 1.9%	98.1% / 1.9%	97.1% / 2.9%	89.2% / 10.8%	91.9% / 8.1%	96.4% / 3.6%

Figure 6(b): Confusion matrix for all classes and all attributes (GoogLeNet)

Conclusion

Presented is a model performance evaluation of pre-trained CNN architectures using transfer learning for fish image classification. The models' performance evaluation for fish image classification by using transfer learning provides the classification accuracy of more than 96% for 18 fish species. The training compared three different CNN models with depth ranging from 5 to 49 convolutional layers. The deepest architecture, ResNet-50, has the smallest gap in the plot of training loss and validation loss plot and requires more training time to achieve the maximum performance. Meanwhile the accuracy gain becomes higher in the test dataset evaluation with the least convolutional layers. AlexNet correctly classified the image as *Rastrelliger kanagurta* with 69.9% probability, ResNet-50 with a 30.0% probability, and GoogLeNet with a 3.5% probability. Both GoogLeNet and ResNet-50 classified the image incorrectly as *Atule mate*. It can be seen that, during validation and test dataset evaluation, the AlexNet model showed a high probability of correctly classifying fish species; while GoogLeNet and ResNet-50 output misclassified the result despite the high classification accuracy. This paper is an extensive study for each of

Confusion Matrix

Target classes (columns, 1–18):
1. Atule mate
2. Caranx ruber
3. Cetoscarus bicolor
4. Decapterus russelli
5. Epinephelus areolatus
6. Epinephelus fasciatus
7. Halichoeres argus
8. Hemiglyphidodon plagiometopon
9. Lutjanus inermis
10. Megalaspis cordyla
11. Monodactylus argenteus
12. Parastromateus niger
13. Pomacentrus alexanderae
14. Rastrelliger kanagurta
15. Salmo salar
16. Scomber scombus
17. Scomberoides commersonnianus
18. Scomberoides tala

Output Class	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	%
Atule mate	513	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	99.6% / 0.4%
Caranx ruber	0	585	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100% / 0.0%
Cetoscarus bicolor	0	0	492	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	100% / 0.0%
Decapterus russelli	2	0	0	288	0	0	0	0	0	1	0	0	0	0	0	0	0	0	99.0% / 1.0%
Epinephelus areolatus	0	0	0	0	421	0	0	0	0	0	0	0	0	0	0	0	0	0	100% / 0.0%
Epinephelus fasciatus	0	0	0	0	0	839	0	0	0	0	0	0	0	1	0	0	0	0	99.9% / 0.1%
Halichoeres argus	0	0	0	0	0	0	540	0	0	0	0	0	0	0	0	0	0	0	100% / 0.0%
Hemiglyphidodon plagiometopon	0	0	0	0	0	0	0	300	0	0	0	0	0	0	0	0	0	0	100% / 0.0%
Lutjanus inermis	0	0	0	0	0	1	0	0	336	1	0	0	0	0	0	0	0	0	99.4% / 0.6%
Megalaspis cordyla	0	0	0	2	0	0	0	0	0	296	0	0	0	1	0	0	2	0	98.3% / 1.7%
Monodactylus argenteus	0	0	0	0	0	0	0	0	0	0	744	1	0	0	0	0	0	0	99.9% / 0.1%
Parastromateus niger	0	0	0	0	0	0	0	0	0	0	0	611	0	0	0	0	0	0	100% / 0.0%
Pomacentrus alexanderae	0	0	0	0	0	0	0	0	0	0	0	0	468	0	0	0	0	0	100% / 0.0%
Rastrelliger kanagurta	6	0	0	9	0	0	0	0	0	2	0	0	0	644	0	1	0	0	97.3% / 2.7%
Salmo salar	0	0	0	0	0	0	0	0	0	0	0	0	0	1	468	0	0	1	99.6% / 0.4%
Scomber scombus	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	479	0	0	99.6% / 0.4%
Scomberoides commersonnianus	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	321	0	100% / 0.0%
Scomberoides tala	5	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	394	98.3% / 1.7%
%	97.2% 2.8%	100% 0.0%	100% 0.0%	96.0% 4.0%	100% 0.0%	99.9% 0.1%	100% 0.0%	100% 0.0%	100% 0.0%	98.7% 1.3%	100% 0.0%	99.8% 0.2%	100% 0.0%	99.4% 0.6%	100% 0.0%	99.8% 0.2%	99.1% 0.9%	99.5% 0.5%	99.5% / 0.5%

Figure 6(c): Confusion matrix for all classes and all attributes (ResNet-50)

the architecture which includes; classification accuracy, learning curve, validation test, top-five prediction and confusion matrix. An accurate fish identification tool is necessary for both experts and non-experts to prevent species misidentification. Transfer learning was able to reduce the training time and improve the performance of deep learning models. These models were able to quickly distinguish the different fish species phenotypes despite a limited quantity of labelled data.

References

Agarap, A. F. (2018). Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*.

Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., ... & Asari, V. K. (2018). The history began

from alexnet: A comprehensive survey on deep learning approaches. *arXiv preprint arXiv:1803.01164*.

Anantharajah, K., Ge, Z., McCool, C., Denman, S., Fookes, C., Corke, P., ... & Sridharan, S. (2014). Local inter-session variability modelling for object classification. In *IEEE Winter Conference on Applications of Computer Vision* (pp. 309-316). IEEE.

Ayob, A. F., Khairuddin, K., Mustafah, Y. M., Salisa, A. R., & Kadir, K. Analysis of Pruned Neural Networks (MobileNetV2-YOLO v2) for Underwater Object Detection. In *Proceedings of the 11th National Technical Seminar on Unmanned System Technology 2019* (pp. 87-98). Springer, Singapore.

Choi, S. (2015). Fish Identification in Underwater Video with Deep Convolutional Neural Network: SNUMedinfo at LifeCLEF Fish task 2015. In *CLEF (Working Notes)*.

Dangelo, M., & Tennefoss, J. Plankton Species Identification via Convolutional Neural Networks.

Gómez-Ríos, A., Tabik, S., Luengo, J., Shihavuddin, A. S. M., Krawczyk, B., & Herrera, F. (2019). Towards highly accurate coral texture images classification using deep convolutional neural networks and data augmentation. *Expert Systems with Applications*, *118*, 315-328.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

Hernández-Serna, A., & Jiménez-Segura, L. F. (2014). Automatic identification of species with neural networks. *PeerJ*, *2*, e563.

Huang, Z., Pan, Z., & Lei, B. (2017). Transfer learning with deep convolutional neural network for SAR target classification with limited labeled data. *Remote Sensing*, *9*(9), 907.

Kozłowski, M., Górecki, P., & Szczypiński, P. M. (2019). Varietal classification of barley by convolutional neural networks. *Biosystems Engineering*, *184*, 155-165.

Krähenbühl, P., Doersch, C., Donahue, J., & Darrell, T. (2015). Data-dependent initializations of convolutional neural networks. *arXiv preprint arXiv:1511.06856*.

Kratzert, F., & Mader, H. (2018). Fish species classification in underwater video monitoring using Convolutional Neural Networks.

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).

Liawatimena, S., Heryadi, Y., Trisetyarso, A., Wibowo, A., Abbas, B. S., & Barlian, E. (2018). A Fish Classification on Images using Transfer Learning and Matlab. In *2018 Indonesian Association for Pattern Recognition International Conference (INAPR)* (pp. 108-112). IEEE.

Liu, Z. (2020). Soft-shell shrimp recognition based on an improved AlexNet for quality evaluations. *Journal of Food Engineering*, *266*, 109698.

Man, M., Abdullah, N., Rahim, M. S. M., & Amin, I. M. (2016). Fish Length Measurement: The Results from Different Types of Digital Camera. *Journal of Advanced Agricultural Technologies Vol*, *3*(1), 67-71.

Nadarajah, P. (1980). Species of Endogonaceae and mycorrhizal association of Elaeis guineensis and Theobroma cacao. *Species of Endogonaceae and mycorrhizal association of Elaeis guineensis and Theobroma cacao.*, 232-237.

Polzounov, A., Terpugova, I., Skiparis, D., & Mihai, A. (2016). Right whale recognition using convolutional neural networks. *arXiv preprint arXiv:1604.05605*.

Rathi, D., Jain, S., & Indu, S. (2017). Underwater fish species classification

using convolutional neural network and deep learning. In *2017 Ninth International Conference on Advances in Pattern Recognition (ICAPR)* (pp. 1-6). IEEE.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., ... & Berg, A. C. (2015). Imagenet large scale visual recognition challenge. *International journal of computer vision*, *115*(3), 211-252.

Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PloS one*, *10*(3), e0118432.

Salman, A., Jalal, A., Shafait, F., Mian, A., Shortis, M., Seager, J., & Harvey, E. (2016). Fish species classification in unconstrained underwater environments based on deep learning. *Limnology and Oceanography: Methods*, *14*(9), 570-585.

Santra, A. K., & Christy, C. J. (2012). Genetic algorithm and confusion matrix for document clustering. *International Journal of Computer Science Issues (IJCSI)*, *9*(1), 322.

Siddiqui, S. A., Salman, A., Malik, M. I., Shafait, F., Mian, A., Shortis, M. R., & Harvey, E. S. (2018). Automatic fish species classification in underwater videos: exploiting pre-trained deep neural network models to compensate for limited labelled data. *ICES Journal of Marine Science*, *75*(1), 374-389.

Springenberg, J. T., Dosovitskiy, A., Brox, T., & Riedmiller, M. (2014). Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*.

Steinbrener, J., Posch, K., & Leitner, R.

(2019). Hyperspectral fruit and vegetable classification using convolutional neural networks. *Computers and Electronics in Agriculture*, *162*, 364-372.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... & Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1-9).

Thenmozhi, K., & Reddy, U. S. (2019). Crop pest classification based on deep convolutional neural network and transfer learning. *Computers and Electronics in Agriculture*, *164*, 104906.

Tyagi, G., Patel, N., & Sethi, I. (2018). A Fine-Tuned Convolution Neural Network Based Approach For Phenotype Classification Of Zebrafish Embryo. *Procedia Computer Science*, *126*, 1138-1144.

Verma, G. K., & Gupta, P. (2018). Wild animal detection using deep convolutional neural network. In *Proceedings of 2nd international conference on computer vision & image processing* (pp. 327-338). Springer, Singapore.

Wang, S., Liu, H., Han, Y., Chen, J., Pan, Y., Cao, Y., ... & Zheng, Y. (2018). Low-altitude remote sensing based on convolutional neural network for weed classification in ecological irrigation area. *IFAC-PapersOnLine*, *51*(17), 298-303.

Xu, J. L., & Sun, D. W. (2018). Computer vision detection of salmon muscle gaping using convolutional neural network features. *Food analytical methods*, *11*(1), 34-47.