Classification of Fish Species with Augmented Data using Deep Convolutional Neural Network

Francis Jesmar P. Montalbo
[1]Technological Institute of the Philippines & [2]Batangas State University
[1]Manila & [2]Batangas City, Philippines
francismontalbo@yahoo.com

Alexander A. Hernandez
Technological Institute of the Philippines
Manila, Philippines
alexander.hernandez@tip.edu.ph

***Abstract*—Classification is one of the primary data science tasks involving large datasets. To date, fish species classification in the Philippines is considered significant for further aquaculture and protection. However, immense efforts and knowledge are necessary to determine fish characteristics through classification. The VGG16 network is one of the top pre-trained models but is still not able to accurately classify common fish species found in Verde Island. This study primarily aims to classify Verde Island fish species using a modified VGG16 network. The VGG16 Deep Convolutional Neural Network (DCNN) undergoes retraining, fine-tuning, and optimization to provide better accuracies in classifying specific Verde Island fish species. Also, this research generated augmented synthetic data for training and testing the model, as there are limited images available. Augmented images are flipped, rotated, cropped, zoomed, and sheared to provide a robust number of features for classification. Results of the training the model achieves 99 percent accuracy for the three different fish species. Hence, this study concludes that a pre-trained model like VGG16 can still improve by fine-tuning, optimization, and data augmentation to classify specific fish species. This paper also includes possible future works determined by the authors.**

Keywords*—*fish classification, deep convolutional neural networks, VGG16, optimization, fine-tuning, data augmentation.

I. Introduction

Verde island passage is home to the most diverse and active ecosystem in the Philippines. The soon to be World Heritage Site provides research opportunities for Several Marine Scientists. Renowned as the "Center of the center of marine shore fish biodiversity in the world" can be located in the province of Batangas. The passage is a strait that stretches from the shores of Mindoro, Romblon, and Marinduque island. Verde Island is considered a sanctuary for many researchers and other experts in the field of marine science and aquaculture. The passage consists of an abundant number of corals, reef formations, rock canyons and other marine life resources numbering to about 600. Also, the Verde Island passage contains about 60% of the different shore fish species in the world, igniting a vast interest in shore-fish research [1][2][3][4]. The massive number of fishes found in the Verde island passage can significantly assist data science experts and marine science researchers to understand the fish ecosystem better.

Collection of fish data on marine science and biodiversity is significantly needed to produce new knowledge. However, data collection is expensive tasks for academics and industry practice [5]. More recently, fish species recognition is considered labor intensive for many experts [6]. Also, the massive amounts of fish species make it more difficult for experts to identify fishes distinctively [7]. Prior studies suggest that identification of specifies can be difficult for non-experts and even for experts. Thus, the correct identification of species is essential in recording data to put forward marine science and aquaculture research [8]. Classification and recognition tasks are highly challenging, especially with a robust number of classes.

More recently, the convolutional neural network (CNN) architecture became famous for image processing, from the straightforward identification of handwriting to a more sophisticated recognition of people crossing pedestrian lanes and human behavior. Earlier versions of CNN are leNet-5 in 1998 by LeCun, AlexNet in 2012 that won as the best classifier based on ImageNet. CNN continues to evolve as more and more research studies are being done to produce better results. The concept of CNN should not be isolated in the field of computer vision as it can branch further into different areas of learning to reach a more intelligent A.I. [9] [10].

Several research studies have used CNN concepts in classifying fish species. Rathi et al. developed a novel method based on CNN to classify 21 species of fishes and achieving an accuracy of 96.29 percent, which is higher compared to the other related studies. A total number of 27,142 of RGB images used together with noise reduction, grayscale, and other relevant techniques to enhance the classification results [11]. Another study by Khalifa et al. used a simplified AlexNet to identify multiple species of fishes. The data used was composed of eight species, with 191 sub-species trained. The results achieved 85.59 percent accuracy and 85.41 percent using AlexNet [12]. Today, other models like the VGG16 and VGG19 by K. Simonyan and A. Zisserman 2014 are one of the top CNN models used for classification as it integrates better learning capabilities compared with AlexNet [13]. In the study of Kratzert el al., VGG16 classified fishes underwater and yielded an accuracy of 93% [14]. Mingwang also used VGG16 with dropout and batch normalization techniques to improve fish classification and achieves 97% accuracy [15]. Santos et al. compared results of VGG16 and VGG19 in classifying fish images and had VGG19 with 83%, which is 2% higher than VGG16 of 81% [16].

The VGG19 may result in higher accuracy, but VGG16 is considerably more scalable as additional layers contribute to the change of results in VGG16. VGG19 did not have much improvement in results with additional layers and just consumed more computing resource. VGG16 had lower

978-1-7281-0758-5/19/$31.00 ©2019 IEEE

errors in terms of classification but with slightly lower accuracy than VGG19 [17]. The VGG16 is considerably better for this study due to its scalability and lower resource consumption.

Hence, this study aims to classify fish species using a modified VGG16 network. The VGG16 is modified by adding Fully Connected (FC) layers via fine-tuning and is optimized. Common Verde Island fish images are collected in open databases and are augmented. The model reliability will also be tested using the performance metrics. This paper aims to contribute by providing an initial frame to classify fish species in the Philippines.

The following provides the research outline presented in a rightly ordered manner: Section 2 tackles the methodology that consists of data acquisition, image augmentation, modeling, fine-tuning, optimization, and training. Section 3 discusses the experimental results of the research study, conclusions, and future works identified by the authors. Lastly, Section 4 contains the relevant references used to accomplish this study.

II. RELATED LITERATURE

A. Artificial Neural Networks

The main idea of Artificial Neural Networks (ANN) was first released in 1943 by McCulloch and Pitts. ANN was constructed based on the operation of the human brain and later evolved to a deep neural network [30]. The CNN architecture rooted in the deep neural network (DNN) composed of multiple different layers. The CNN architecture based on the concept DNN except, CNN uses a convolution and pooling layer separated by hidden layers. CNN sets itself apart from the DNN as it uses a shared kernel rather than a separate kernel. The ANN structure that started DNN and CNN is made up of the input, hidden, and output layers together with the weights and activation functions represented in figure [18] [19]. Figure 1 illustrates the M-P ANN model.

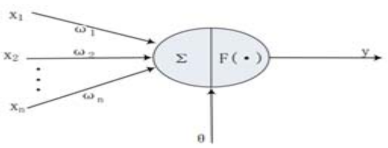

Figure 1: M-P ANN Model (Cheng, 2016)

The illustrated formula represents the ANN variable x_i is the number of inputs and w_i as the corresponding weight, θ represents the threshold and f as the activation function. Figure 2 provides a detailed structure of the ANN [18].

$$y = f(\sum x_i \omega_i - \theta)$$

Equation 1. M-P Model Notation

B. Convolutional Neural Network

The architecture of CNN is a deep neural network made up of an input layer, convolution layer, pooling layer, fully connected layer, and output layer [28] [29]. The image enters the input layer and is turned into a 32x32x3 matrix with 32x32 as the dimensions and three as the color scheme RGB. The image matrix is then entered to a convolution layer to extract the features of the image by using a filter and sliding it over the image to generate the weights called a feature map. Next is entering the pooling layer, each feature map undergoes the pooling layer. The image space is reduced into a smaller dimension to lessen its parameters and leaving only the critical features. Fully connected layers are considered the model classifier, and the output from the previous layers is used to determine the best feature fit for each class. Using an activation function fully connected layers convert the vector values into elements of 0-1 and 1 [31]. The image is received by the output layer in a vector format predicting the classification from the input layer [9]. Figure 2 shows the architecture of CNN.

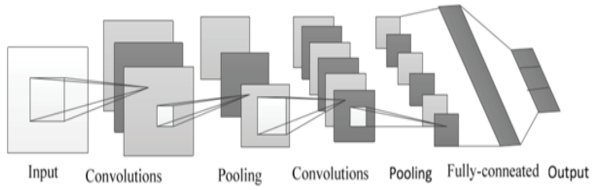

Figure 2: Convolutional Neural Network

C. VGG16 Model

The VGG16 model is a modified architecture based on the CNN model developed by K. Simonyan and A. Zisserman. VGG16 is one of the most popular entries on the ImageNet 2014 competition, achieving an accuracy score of 92.7% with 14 million sets of images with 1000 classifications [13]. The results are promising with high potential for further improvements. Over time, deep CNN likeVGG16 model significantly improved for classification activities. Furthermore, depth was considered the main factor in improving model performance. The model is composed of 16-layers with an input layer of 224x224 RGB and having only a 3x3 convolutional layer throughout the whole model. The VGG16 reduce the input images in size through the max-pooling layers. Also, the model is composed of 138 million parameters. Training such a large model would take an extended period, considering that the depth of the model is deeper compared with other previous CNN models [20]. Figure 3 provides an overview of the VGG16 architecture.

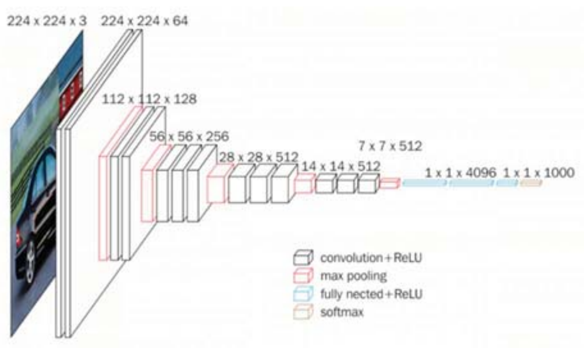

Figure 3: VGG16 Model (Hassan, 2018)

III. MATERIALS AND METHODS

The main objective of this study, as shown in figure 4, is to create a fish classifier model. The classifier model uses the pre-trained VGG16 model that will be retrained to fit better features of the Verde Island fish species. The model feeds more on the augmented fish images due to the limited amount of data available. The VGG16 fine-tune process adds four additional FC layers to capture new features of the three fish species. The modified model is then trained using the augmented images together with a new set of features. The reliability of the model is tested using the performance metrics. The results are then illustrated using a confusion matrix to provide depth and better analysis. The actions mentioned are explained more in-depth in the following subsections.

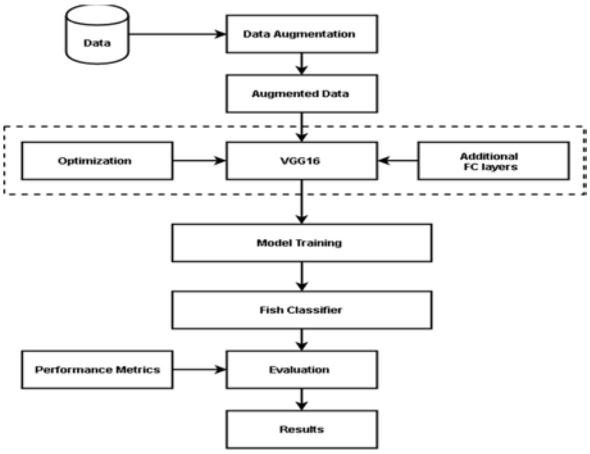

Figure 4: Fish Classifier Methodology

A. Data Preparation

This study used images of commonly found fish species in the Verde Island Passage. The following fish images were verified to determine its integrity as misclassification of fish species may cause unwanted results. The dataset is composed of three fish species specified in Table 1. Table 1 presents the number of images with a total of 530, which were used for training and testing the model. The model used 455 images for training and 75 images for testing. Following images displayed in Table 1 are taken from FishBase. The following FishBase images used are under the creative commons license that allows the free use of content. Additional Images populated came from data augmentation.

Table 1: Fish Species Dataset

Fish Family Species	Sample Image	Train	Test	Source
Amphiprion clarkii (AC)		140	10	Randall, J.E., 1997
Chaetodon baronessa (ChB)		150	40	
Ctenochaetus binotatus (CtB)		165	25	
Total	530	455	75	

B. Data Augmentation

Most domains today still have the scarcity and inability to access big data. With limited data available, a solution of data augmentation solved the problem. In this study, transformation and generation of new versions of the original image data resulted in an increased number of images from the original dataset [21]. Data augmentation is also a way to prevent or reduce overfitting of the model by increasing the training data [22]. This study used traditional transformation, a technique of image transformation and color modification, and the distortion of images [23]. The newly produced data from augmentation is used in the model to increase the number of training data for higher accuracy of classification. Figure 5 presents the samples of augmented images contained in the dataset.

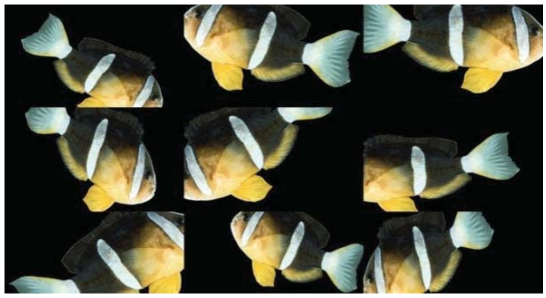

Figure 5: Augmented Data Samples (FishBase – Randall, 1997)

C. Modeling and Fine Tuning

A technique based on Transfer Learning [24], this method allows the pre-trained model VGG16 to support the classification of the images using the dataset. The weights of ImageNet transferred to the model for the training of the classifier. The initial step for fine-tuning is freezing the lower layers and adding FC layers using only a minimal learning rate. The main reason for freezing the lower layers is to capture a standard set of features like edges, curves, patterns, and more. The FC layers are replaced based on the classification task needed, as the original layers are too large due to its original purpose. The layers are tweaked accordingly to the amount of data and classes collected. Finally, an amount of 0.001 learning rate is used to prevent any immediate distortion of the weights that can result in a lower accuracy result. Figure 6 presents a summary of the fine-tuned pre-trained model.

Layer (type)	Output Shape	Param #
input_2 (InputLayer)	(None, 224, 224, 3)	0
block1_conv1 (Conv2D)	(None, 224, 224, 64)	1792
block1_conv2 (Conv2D)	(None, 224, 224, 64)	36928
block1_pool (MaxPooling2D)	(None, 112, 112, 64)	0
block2_conv1 (Conv2D)	(None, 112, 112, 128)	73856
block2_conv2 (Conv2D)	(None, 112, 112, 128)	147584
block2_pool (MaxPooling2D)	(None, 56, 56, 128)	0
block3_conv1 (Conv2D)	(None, 56, 56, 256)	295168
block3_conv2 (Conv2D)	(None, 56, 56, 256)	590080
block3_conv3 (Conv2D)	(None, 56, 56, 256)	590080
block3_pool (MaxPooling2D)	(None, 28, 28, 256)	0
block4_conv1 (Conv2D)	(None, 28, 28, 512)	1180160
block4_conv2 (Conv2D)	(None, 28, 28, 512)	2359808
block4_conv3 (Conv2D)	(None, 28, 28, 512)	2359808
block4_pool (MaxPooling2D)	(None, 14, 14, 512)	0
block5_conv1 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv2 (Conv2D)	(None, 14, 14, 512)	2359808
block5_conv3 (Conv2D)	(None, 14, 14, 512)	2359808
block5_pool (MaxPooling2D)	(None, 7, 7, 512)	0
global_average_pooling2d_2 (	(None, 512)	0
dense_5 (Dense)	(None, 1024)	525312
dense_6 (Dense)	(None, 1024)	1049600
dense_7 (Dense)	(None, 512)	524800
dense_8 (Dense)	(None, 3)	1539

Total params: 16,815,939
Trainable params: 2,101,251
Non-trainable params: 14,714,688

Figure 6: Summary of Fine-Tuned Pre-Trained Model

D. Optimization

Due to the default weights from the pre-trained model not being designed for the proposed tasks, a higher loss function can occur. An Optimization method, like RMSProp, is used to aid this problem. The optimizer can reduce the amount of cost or loss function. The cost function is reduced by continuously adjusting the set of values for the weight parameters until it reaches its optimized form for the desired tasks. Lowering the cost function would highly benefit the accuracy of the model by having a lesser difference of the actual output and predicted output with the new set of weights. Several optimizers are available with the help of Keras libraries. The RMS Prop or Root Mean Squared Propagation algorithm originated from a lecture by Geoffrey Hinton [25]. This optimization algorithm came from the Gradient Descent Algorithm with the combined concepts of the Adagrad and Adadelta methods, being a solution to the diminishing learning rate of Adagrad [26]. RMS Prop commonly aids in deep neural networks and computer vision as an optimizer [27]. In this experiment, the RMSProp remolded the parameters of the modified model to adapt to the previous weights of the original VGG16 model. The modified model improved in detecting better features of the three fish species, generating higher accuracy of predictions, as shown in Table 2. Equation 2 provides the equation of RMSProp optimization.

$$MeanSquare(w,t) = 0.9\ MeanSquare(w,\ t{-}1) + 0.1\left(\partial E/_{\partial w}(t)\right)^2$$

Equation 2: RMSProp Notation (Hinton, 2012)

E. Model Training

Training or fitting the model, a batch size of 16 with ten epochs was assigned based on the current RAM of the machine. Also, an early stopping function was included to eliminate unnecessary consumption of training time and computing resources that does not benefit the final results of the model. The training process reached the tenth epoch with a validation loss of only 0.0568. The researchers also used a checkpoint to record the best training results. With the callback functions of the Keras library, the checkpoint is created to prevent any unwanted overwrite on the best training weights when another compilation happens.

F. Evaluation of Fish Classifier

Performance metrics are used to define the capability of the model. The following metrics define the accuracy, recall, precision, F1 score, and Confusion matrix of the model in classifying the images.

Accuracy entails the correct and incorrect predictions of the model. The accuracy relies on the number True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN). To compute for the accuracy, the TP and TN are added and divided by the sum of TP, TN, FP, and FN. Equation 3 provides the notation to compute for the accuracy.

$$Accuracy = \frac{TP+TN}{P+N} = \frac{TP+TN}{TP+TN+FP+FN}$$

Equation 3: Accuracy Formula

Recall or also known as the True Positive Rate (TPR) determines the total ratio of the correct classification of positive results on the model. The model has a high percentage of classifying correctly if the recall produces a high value of recall based on a low number of FN. Recall can be denoted in the following formula, as presented in Equation 4.

$$Recall = \frac{TP}{TP+FN}$$

Equation 4: Recall Formula

Precision or Positive Predictive Value (PPV) determines the exact correct prediction or classification. The numbers of FP's are considered low if the precision rate is high, which is suitable for a model. Precision can be computed as follows, as presented in Equation 5.

$$Precision = \frac{TP}{TP+FP}$$

Equation 5: Precision Formula

F_1Score provides a balance between Precision and Recall. The F_1-score is different from the accuracy as it focuses more on the FP and FN, while accuracy focuses more on the TP. The F_1-score can be computed in the following notation, as presented in Equation 6.

$$F_1 = 2\left(\frac{PPV * TPR}{PPV + TPR}\right)$$

Equation 6: F_1-Score Formula

IV. EXPERIMENTAL RESULTS

This section presents the results of the conducted experiment. The fine-tuned and optimized VGG16 model was evaluated based on the performance metrics, training results, and reliability. A total of 530 fish images separated into three different fish species with a different number of training images and test images is applied. Amphiprion clarkii (AC) had 140 training images, and 10 test images, Chaetodon baronessa (ChB) had 150 and 40, Ctenochaetus binotatus (CtB) had 165 and 25, respectively.

Table 2: Model Training Results

Epoch	Accuracy	Loss	Validation Loss	Validation Accuracy
1	0.5282	1.2062	0.6230	0.7288
2	0.8101	0.5330	0.2728	0.8375
3	0.9040	0.2907	0.4482	0.8305
4	0.9442	**0.1773**	0.0997	0.9322
5	0.9352	0.2379	0.4512	0.8814
6	0.9464	0.1764	0.2477	0.9322
7	0.9620	0.1927	0.0980	0.9688
8	0.9576	0.1951	0.0663	0.9492
9	0.9263	0.2382	0.0859	0.9661
10	**0.9587**	0.2495	**0.0568**	**1.0000**

Table 2 indicates the number of epochs used to train the model. The increase of accuracy per each epoch also shows a decrease in the loss. The results of the accuracy increasing and loss decreasing are significantly parallel to the validation accuracy and validation loss. Throughout the training process, it is noticeable that the accuracy in epochs five (93%), eight (95%), and nine (92%) had decreased from their previous epochs of 94%, 96%, and 96%. The loss also had an inconsistent change on its results similar to the validation loss, while validation accuracy had inconsistent increase from epoch four (93%) to five (88%), and seven (97%) to eight (95%). This behavior significantly occurs in unbalanced datasets and is a sign of overfitting. However, the use of the early stopping function from Keras eradicates this problem by saving only the best-fitted model. The line graph illustrated in figure 7 shows an overview of the model fit using the values in Table 2 to justify the model reliability.

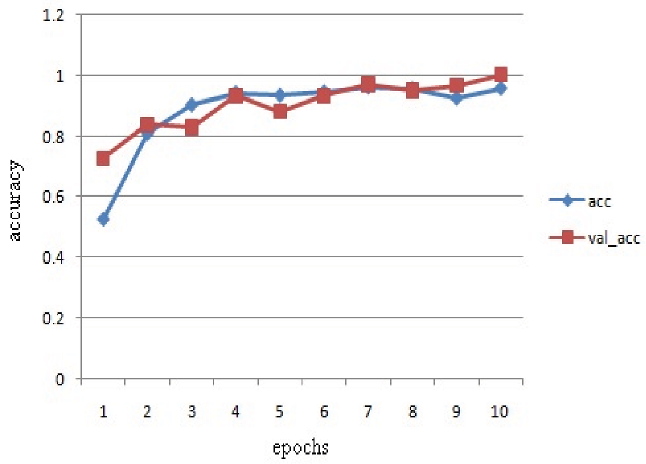

Figure 7: Model Reliability Results

The line graph in Figure 7 shows certain events of overfitting. However, the model was still able to control the accuracy fit from the validation accuracy well. Epochs one, three, five, nine, and ten had a bit of over and undershooting of the lines indicating that the model can misclassify anytime. Even with the results provided in Figure 13, the model is still considered reliable in terms of classification as the model only had a short distance of over and undershooting. The chance of misclassification is significantly low and can be further analyzed using the performance metrics and confusion matrix in Table 3 and Figure 8.

Table 3: Model Testing Performance Results

Species	Accuracy	Precision	Recall	F1-score
Amphiprion Clarkii (AC)	100%	100%	88%	100%
Chaetodon baronessa (ChB)	98.67%	100%	98%	99%
Ctenochaetus binotatus (CtB)	98.67%	96%	100%	98%

Table 3 consists of the performance results of the model in terms of accuracy, precision, recall, and F1-score. The results provide an overview of how reliable and accurate the model can classify all three species apart. AC achieved the highest accuracy (100%), followed by ChB and CtB (98.67%). The total True Positives identified was 74, with an overall accuracy of 98.67%. The yielded results are relatively high, but still, have a chance of 1.33% misclassification. However, this may not entirely affect the classifier and can improve in future applications.

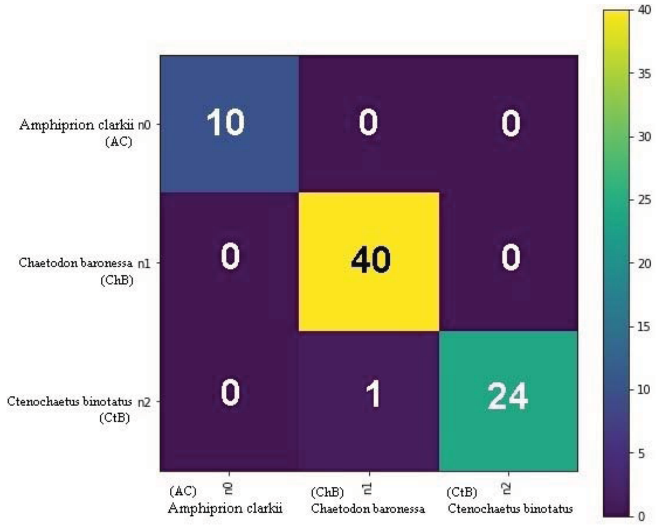

Figure 8: Confusion Matrix

The confusion matrix in figure 8 is a table of values used to represent the performance of a model, usually in terms of classification. In the illustrated confusion matrix, the highest number of validation data was the ChB (40) next to CtB (25) and the lowest as AC (10).

The first panel on the left of figure 8 is the total number of correctly classified species of AC. The results of AC had a perfect 10 out of 10 classifications with the test data indicating 100% accuracy. ChB similarly received a perfect score with a 40 out of 40 correct classifications. However, CtB did not fair much compared to the others due to a single error. The total score yielded by the confusion matrix is 74 out of 75, with one error found in CtB. With the results indicated, the over-all model accuracy is 99%.

V. Conclusion

The use of previously trained models provided immense benefits for research and development. Transfer learning provides less work in training large models to perform a specific classification intensively. The VGG16 model is one of the most promising models developed. However, still was not able to classify images of fish species from Verde Island correctly. The problem in classification was solved by shifting the weights of the model by training it based on the new dataset for the needed classification task. The newly trained model from the previous VGG16 model had shown significant results compared to its prior weights. Even with the low number of images used, the dataset was populated further by the application of image augmentation. The populated images provided a tremendous increase in the final results of classification. To further enhance the accuracy, the use of RMSProp optimization decreased the loss of values during the training process. Fine-tuning methods are applied to add depth to the network that highly contributed to the final results.

The model reached an overall accuracy of 99%. The final results classified ten images of AC with 100% precision using 140 training data. 150 images of ChB were used for training and validated using 40 images and manage to reach 98.67%. CtB had 165 training data and 25 for validation generating a result of 98.67%. The results were positively good, though there is a minimal amount of overfitting along with the training. The following results indicate that even with a small-sized dataset, the application of transfer learning, fine-tuning, optimization, can significantly enhance the desired classification task even with a small amount of data.

This study is not free from limitations. This study could be further extended by developing web and mobile application that integrates the model to classify fish species and species that have not yet existed on the dataset. The future application can also be used to assist researchers to classify better with less needed work. Also, this study suggests collecting a large number of a dataset to further train and test the model.

REFERENCES

[1] "Verde Island Passage," Verde Island Passage. [Online]. Available: https://www.conservation.org/philippines/projects/verde-island-passage. [Accessed: July 8, 2019]

[2] S. Pa-a, "Declaration of Verde Island Passage as World Heritage Site sought," Philippine News Agency RSS, 24-Sep-2018. [Online]. Available: https://www.pna.gov.ph/articles/1048958. [Accessed: July 8, 2019].

[3] V. Horigue and W. Y. Licuanan, "Sensitivities of coral reefs in the Verde Island passage to climate change," *Journal of Coral Reef Studies,* vol. 15, pp. 359-365, 2013.

[4] K. E. Carpenter and V. G. Springer, "The center of the center of marine shore fish biodiversity: the Philippine Islands," *Environmental Biology of Fishes*, vol. 72, no. 4, pp. 467–480, 2005.

[5] X. Yang, Z. Zeng, S. G. Teo, L. Wang, V. Chandrasekhar, and S. Hoi, "Deep Learning for Practical Image Recognition," Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining - KDD 18, 2018, New York, NY, USA,2018, *pp.923-931.* doi:10.1145/3219819.321990

[6] H. Qin, X. Li, J. Liang, Y. Peng, and C. Zhang, "DeepFish: Accurate underwater live fish recognition with a deep architecture," *Neurocomputing*, vol. 187, Apr., pp. 49-58, 2016.

[7] M. M. M. Fouad, H. M. Zawbaa, N. El-Bendary and A. E. Hassanien, "Automatic Nile Tilapia fish classification approach using machine learning techniques," *13th International Conference on Hybrid Intelligent Systems (HIS 2013)*, Gammarth, 2013, pp. 173-178.doi: 10.1109/HIS.2013.6920477

[8] G. E. Austen, M. Bindemann, R. A. Griffiths, and D. L. Roberts, "Species identification by experts and non-experts: comparing images from field guides," *Scientific Reports,* vol. 6, no. 1, Sept., 2016

[9] Q. Zhang, "Convolutional Neural Networks," In *Proceedings of the 3rd International Conference on Electromechanical Control Technology and Transportation*, 2018, pp. 434-439.

[10] G. Ding *et al*., "Fish recognition using convolutional neural network," *OCEANS 2017 - Anchorage*, Anchorage, AK, 2017, pp. 1-4.

[11] D. Rathi, S. Jain and S. Indu, "Underwater Fish Species Classification using Convolutional Neural Network and Deep Learning," *2017 Ninth International Conference on Advances in Pattern Recognition (ICAPR)*, Bangalore, 2017, pp. 1-6. doi: 10.1109/ICAPR.2017.8593044

[12] N. E .M. Khalifa, M. H. N Taha, and A. E. Hassanien, "Aquarium Family Fish Species Identification System Using Deep Neural Networks," *Advances in Intelligent Systems and Computing Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2018*, vol. 845, pp. 347-356.doi: 10.1007/978-3-319-99010-1_32

[13] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv:1409.1556, 2014

[14] F. Kratzert and H. Mader, "Fish species classification in underwater video monitoring using Convolutional Neural Networks," 2018.doi: 10.31223/osf.io/dxwtz

[15] L. Mingwang, "Fish Image Recognition and Separation Based on Convolutional Neural Network[J]". *Image and Multimedia Technology 2017,*vol. 3, pp. 82-83.

[16] A. A. D. Santos and W. N. Gonçalves, "Improving Pantanal fish species recognition through taxonomic ranks in convolutional neural networks," *Ecological Informatics*, vol. 53, p. 100977, 2019.doi: 10.1016/j.ecoinf.2019.100977

[17] S. Bianco, R. Cadene, L. Celona and P. Napoletano, "Benchmark Analysis of Representative Deep Neural Network Architectures," in *IEEE Access*, vol. 6, pp. 64270-64277, 2018. doi: 10.1109/ACCESS.2018.2877890

[18] T. Cheng, P. Wen, and Y. Li, "Research Status of Artificial Neural Network and Its Application Assumption in Aviation," *2016 12th International Conference on Computational Intelligence and Security (CIS)*, Wuxi, 2016, pp. 407-410.doi: 10.1109/CIS.2016.0099

[19] W. S. McCulloch and W. Pitts "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol.5, no. 4, pp.115-133, 1943.doi: 10.1007/bf02478259

[20] S. Liu and W. Deng, "Very deep convolutional neural network based image classification using small training sample size," *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, Kuala Lumpur, 2015, pp. 730-734. doi: 10.1109/ACPR.2015.7486599

[21] C. Shorten and T.M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," *Journal of Big Data,* vol. 6, no.1, p. 60,Jun, 2019.doi: 10.1186/s40537-019-0197-0

[22] J. Wang and L. Perez, "The effectiveness of data augmentation in image classification using deep learning," arXiv: 1712.04621, 2017

[23] A. Mikołajczyk and M. Grochowski, "Data augmentation for improving deep learning in image classification problem," *2018 International Interdisciplinary PhD Workshop (IIPhDW)*, Swinoujście, 2018, pp. 117-122.doi: 10.1109/IIPHDW.2018.8388338

[24] C. Tan, F. Sun, W. Zhang, C. Yang, and C. Liu, "A Survey on Deep Transfer Learning," *Artificial Neural Networks and Machine Learning – ICANN 2018. ICANN 2018. Lecture Notes in Computer Science,* vol. 11141, pp. 270-279, Sept. 2018,doi: 10.1007/978-3-030-01424-7_27

[25] T.Tielman and G. Hinton. Class Lecture, Topic: "6.5-rmsprop: Divide the gradient by a running average of its recent magnitude." COURSERA: Neural networks for machine learning 4, no. 2, 2012

[26] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv:1609.04747*, 2016.

[27] M. C. Mukkamala and M. Hein, "Variants of RMSProp and Adagrad with Logarithmic Regret Bounds," *Proceedings of the 34th International Conference on Machine Learning - Volume 70 (ICML'17)*, Sydney, NSW, Australia, 2017, pp. 2545-2553.

[28] Alimboyong, C. R., & Hernandez, A. A. (2019, March). An Improved Deep Neural Network for Classification of Plant Seedling Images. In *2019 IEEE 15th International Colloquium on Signal Processing & Its Applications (CSPA)* (pp. 217-222). IEEE.

[29] Alimboyong, C. R., Hernandez, A. A., & Medina, R. P. (2018, October). Classification of Plant Seedling Images Using Deep Learning. In *TENCON 2018-2018 IEEE Region 10 Conference*(pp. 1839-1844). IEEE.

[30] Malinao, R. M. L., & Hernandez, A. A. Artocarpus Trees Classification using Convolutional Neural Network. In 2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM) (pp. 1-6). IEEE.

[31] Malinao, R. M. L., & Hernandez, A. A. (2018, June). Classifying Breadfruit Tree using Artificial Neural Networks. In *Proceedings of the 6th ACM/ACIS International Conference on Applied Computing and Information Technology* (pp. 27-31). ACM.