Developing a Classification CNN Model to Classify Different Types of Fish

Chapter · October 2022

DOI: 10.1007/978-3-031-19958-5_50

6 authors, including:

Farhan Ahmed
East West University (Bangladesh)

Bijoy Basak
East West University (Bangladesh)

Simonta Chakraborty
East West University (Bangladesh)

Ahmed Wasif Reza
East West University (Bangladesh)

Developing a Classification CNN Model to Classify Different Types of Fish

Farhan Ahmed[1], Bijoy Basak[1], Simonta Chakraborty[1], Tumpa Karmokar[1], Ahmed Wasif Reza[1], Omar Tawhid Imam[2], and Mohammad Shamsul Arefin[3,4](✉)

[1] Department of Computer Science and Engineering, East West University, Dhaka 1212, Bangladesh
wasif@ewubd.edu
[2] Department of EEE, Bangladesh University of Engineering and Technology, Dhaka, Bangladesh
[3] Department of Computer Science and Engineering, Daffodil International University, Dhaka 1341, Bangladesh
sarefin@cuet.ac.bd
[4] Department of Computer Science and Engineering, Chittagong University of Engineering and Technology, Chattogram 4349, Bangladesh

Abstract. Identifying any fish type can be difficult for people who are not familiar with fish. Implementation of a fish classification machine learning model can become helpful in this scope. The purpose of this paper is to build such a fish classification machine learning model. With this classification model, people will be able to identify the class or type of fish even without much experience with fish. Different types of fish have different nutrition, vitamin, and fat content. Thus, this model can be helpful to ensure better nutrition intake as well. As we have to classify types of fish, we implemented a Convolutional Neural Network (CNN) with Keras along with a modified VGG16 transfer learning model. With the CNN model, the accuracy of our training is 96.67%, and classification accuracy with the modified VGG16 is 97.44%. For validation, with the CNN model, accuracy is 99.92%, and classification accuracy with the VGG16 is 99.76%.

Keywords: Fish dataset · Feature extraction · VGG16 · Machine learning · Transfer learning · Image augmentation · Image classification

1 Introduction

The process of identifying fish species based on their features is called fish recognition [1]. Also, can be stated as a process in which targeted fish species are identified based on the similarities of images of representative specimens [1]. In simpler words, identifying fish species by extracting their features from their images is called fish recognition. Fish are a major source of low-fat high-quality protein, omega-3 fatty acids, and vitamins like D and B2 [2]. But not all fishes provide the same amount of nutrition. So, for a more normalized intake of nutrition consuming diverse types of fish can become healthy.

P. Vasant et al. (Eds.): ICICO 2022, LNNS 569, pp. 1–11, 2023.
https://doi.org/10.1007/978-3-031-19958-5_50

But there is various kind of living fish species in the world, more than 32,000 [3]. This makes the identification of fish an important subject. However, fish classification is more difficult than other object or image classification. The classification of marine life differs from other things such as dogs, flowers, and cats. Fish classification is more difficult because of the diversity of species and similarities within classes, such as textures and shapes. Furthermore, due to light interference, the acquired fish images are frequently reflected too darkly, making image feature extraction difficult.

Regardless of the difficulties, fish identification with image classification can be used variously in the discipline of fish research, fish knowledge popularization, aquaculture, and rare fish conservation. The creation of a comprehensive database consisting of data of various fish species and the use of image classification methods to classify them can help not only to better protect fish resources but also to contribute to marine fishery production development, as well as have research and economic value.

Nowadays object detection using machine learning has been used widely in various fields. There are a lot of techniques used for image classification like SVM, K-Nearest Neighbor (KNN), K-means Clustering, Neural Network, etc. Among the neural network techniques, image classification with Convolutional Neural Network (CNN) is one of the most popular. The ability to affect multiple dimensions of an object's overall scale made image classification with CNN made it successful [4]. Image classification is widely used in face recognition, object recognition, and detecting any specified image. For our implementation, we used two CNN models. One is a pre-trained VGG16 model with transfer learning and the other one is our own CNN made from scratch using TensorFlow and Keras. VGG16 is a famous and widely used pre-trained model. Using two models can also give an insight into how the VGG16 model can perform a fish classification scenario.

2 Related Work

We found a few papers about the works of image processing. They worked on different types of the images such as a flower, fish, trees, fruit, etc.

Siraj et al. [5] used the algorithm of Neural Networks (NN), and Logistic Regression, and the method of this algorithm is SVM. The result of this paper is that it can detect 95% of flowers from the dataset.

Kartika et al. [6], used the K-means algorithm. And in this algorithm, they use the SVM and Naïve Bayes method. For the object separation, they used K-means and they used the Naïve Bayes method for comparison with SVM. They also used k-fold for cross-validation to get better accuracy. The resulting accuracy is 97%.

Hridayami et al. [7] used a deep CNN model VGG16 to detect fish. They used the transfer learning method to keep the models pre-trained with ImageNet weights. Their fish dataset had images of fifty fish species, each category (species) had 15 images, among which 10 were for training and 5 were for testing. In the study, they trained the model on four different types of datasets: RGB color space images, Canny filter images, composite images, and composite images mixed with RGB images. The results show that the blended image of the image mixed with the RGB image training model has the highest true acceptance rate (GAR) value of 96.4%, followed by the RGB color space image training model with a GAR of 92.4%. The trained model with mixed images showed the lowest GAR of 75.6%.

Montalbo et al. [8] use CNN. They applied it to classify Verde Island fish species. They worked with three species of fish. They modified the VGG16 model and used Deep Convolutional Neural Network (DCNN). Their augmented images are filliped, cropped, sheared, zoomed, and rotated. Four additional FC layers are used to capture new features of the fish images. The accuracy of the model in this paper is 99%. In our VGG16 model, we used it differently from them. We apply seven-layer in our model.

Das et al. [9] use the algorithm of pre-trained VGG16. They applied inter-domain transfer learning. They also used Deep Convolutional Neural Network (DCNN). They worked on the RVL-CDOP document image dataset. The accuracy of the model in this paper is 92.21%.

Miyazono et al. [10] developed a technique to recognize fish species with image processing. With their developed technique, the user can be able to identify the poisonous and non-poisonous fish before eating. Fish names and characteristics can also be checked by their developing detection system. For developing their detection model, they used the image normalization technique to resize the images. They used four featured points for annotating the images. By measuring the featured points, they created various channels for developing their CNN network. Then they used the CIFAR-10 network and AlexNet which are two well-known CNN architectures. These two networks are pre-training networks. After that, they collected fifty fish species to use as their dataset. They divided their dataset into four groups and applied 4-fold cross-validation. With their dataset, they found an accuracy of 71.1% for one candidate and 91.4% for five candidates.

Iqbal et al. [11] created an automation and classification system for fish species. Their work, helped marine biologists to better understand fish species and their habitats. Their proposed model is a reduced version of the AlexNet model, which consists of 4 convolutional layers and 2 fully connected layers. They presented a comparison with other deep learning models like VGGNet and AlexNet. Their resulting model (modified AlexNet model) achieved test accuracy of 90.48%, while the original AlexNet model achieved 86.65%.

Desai et al. [12] used the algorithm pre-trained VGG16 model. She took a flower dataset that presents two types of flower data. She used the VGG16-based model for feature extraction. This model used features extracted from the convolutional base model and compiled using the Adam optimizer. In this model, she got 90% accuracy for the flower dataset.

Bird et al. [13] recommended a study and analysis to determine whether CNN architecture works more accurately and efficiently with new datasets. They validated their work using the transfer learning model. Through the use of a transfer learning model, they were able to establish an appropriate architecture for image recognition. They demonstrated a comparison between CIFAR-10 and their suggested model with 500 epochs and 4000 epochs. For 500 epochs, they discovered an average accuracy of 91%, and for 4000 epochs, the accuracy was 96.5% which was better than the CIFAR-10 accuracy of 70.1%.

3 Methodology

3.1 System Architecture and Design

Our first model is a VGG16 transfer learning model. A learner's proficiency in skill or knowledge in one context which enables them to use that same skill or context in another context can be taken as the cognitive practice of "Transfer" [14]. Similarly, for machine learning models, the pre-trained models such as VGG16 are trained with thousands of images, and with transfer learning, they can improve the performance of the model significantly. Making a CNN model from scratch has a risk of overfitting and poor performance due to several reasons like insufficient data, inappropriate parameters, etc.

In our implementation of the VGG16 Transfer learning model (Fig. 1), we loaded the VGG16 model without the top or input layer. As the input layer needs to be replaced during transfer learning to connect the pre-trained model with the new model. By removing the top layer, only the weights for various layers of the pre-trained model are included in the new model which increases feature extraction and efficiency of the model. We attached some more layers to the model to create the fully connected classifier layers.

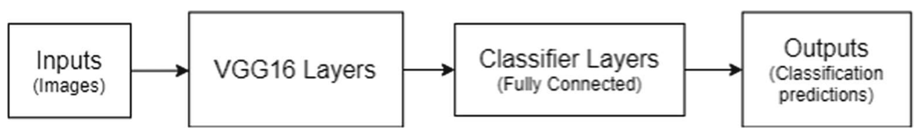

Fig. 1. VGG16 transfer learning model architecture

For the other model (Fig. 2), we made our own Convolutional Neural Network (CNN) model for fish detection with TensorFlow and Keras. The final five layers of the model are kept like what we used for VGG16. So, the other layers such as Convolutional blocks (Conv2d), and the max-pooling layer (MaxPooling2D) will be the ones to create the difference in performance between the two models.

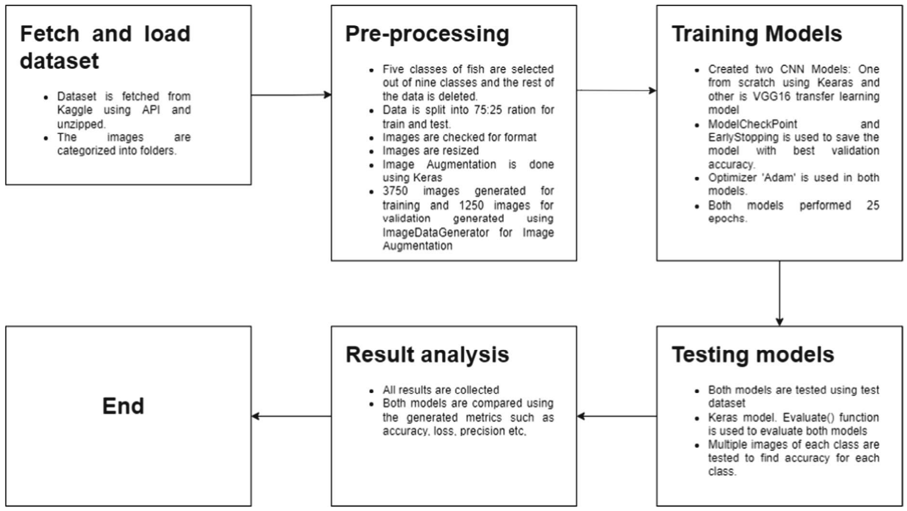

Fig. 2. System architecture of fish classification

3.2 Dataset Description

The dataset is a publicly available dataset collected from Kaggle. The dataset has nine thousand images of nine fish categories. From this, we took only five categories with five thousand images to perform the image classification. The dataset also contains fifty images of each fish category for testing. All the classes or categories of the fishes contain the same number of images for both training and testing. Thus, the dataset is balanced. Black Sea Sprat, Sea Bass, Red Mullet, Glit Head Bream, and Shrimp images from the dataset were used. Some samples of our dataset are presented in Fig. 3.

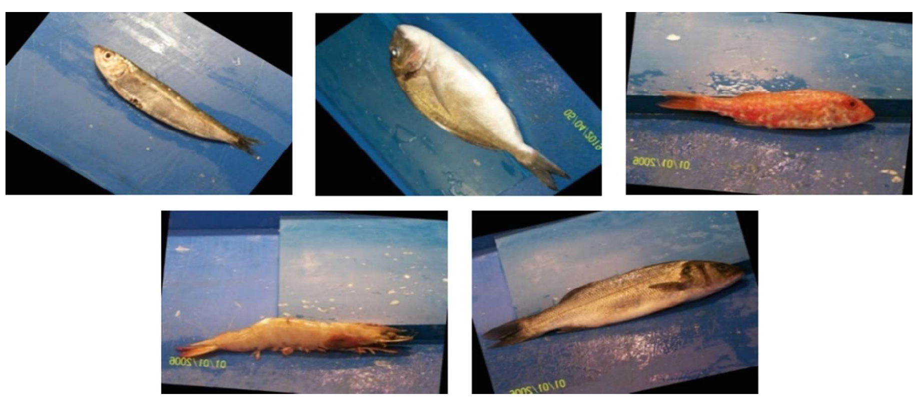

Fig. 3. Images from the dataset

3.3 Data Preprocessing

First, we split the data (images) for training and validation purposes in a 75:25 ratio. 75% of the images were randomly selected for training and the rest were for validation. After the splitting, we had 750 pictures for each five fish classification folders as training data and 250 pictures for each 5 fish classification folders as validation data. For preprocessing of the images, we used the ImageDataGenerator function of the Keras image preprocessing library. Using the function, we rescaled the images by 1/255 of their original scale. Random horizontal flipping was enabled. A random zoom range of 0.4 and random rotation of 30° was used. We resized all the images to 224×224 pixels. As we have used five thousand pictures for the training that can be small for the deep learning model. It creates a scope for an overfitting model. For resolving that issue, we have used image augmentation to augment the images to create more data for the model to learn from.

3.4 Fish Classification

In VGG16 the topmost layer or input layer is removed for transfer learning. The rest of the layers are kept. Five more layers consist of a Flatten layer, a Dropout layer, and three Dense layers. These layers extract information from the provided images.

On the other approach, for the CNN model, the topmost or input layer is a Conv2D layer. Then few more layers consisting of Conv2D, MaxPooling2D, Flatten, Dense, and Dropout are used. These layers extract information from the provided images.

4 Result and Discussion

4.1 Experimental Setup

Windows Operating System: We used a computer with Windows operating system for this project. The system configuration used in this project, Intel(R), Core (TM) i5-8300H, CPU limit: 2.30 GHz 2.30 GHz, GPU: GeForce GTX-1050Ti, Installed RAM capacity: 16.0 GB. The system type is a 64-bit operating system, and the processor is an x64-based processor.

Google Colab: We used Google's Colaboratory for our implementation as it supports TensorFlow and Keras APIs. Also, as our dataset is large, the GPU runtime of the Collaboratory was an ideal option for the implementations.

4.2 Implementation

A stepwise approach to our implementation:

1. Make a training and validation directory
2. Split dataset into train and validation by 75:25 ratio
3. Preprocess images

4. Modify pre-trained model/Build CNN model
5. Train the model with a batch size of 32 and 25 epochs
6. Save the best model
7. Load weights of the best model
8. Evaluate the model using a test dataset
9. Test with images that are not in the dataset
10. Get predictions for the image.

4.3 Performance Evaluation

Comparison values are of the best models of the VGG16 transfer learning model and CNN from the scratch model. The training and validation accuracy of the models are shown in Table 1. The accuracies of both these models are plotted in Figs. 4 and 5.

Table 1. Modified VGG16 vs CNN training and validation accuracy

	Modified VGG16	CNN
Training	97.44	96.67
Validation	99.76	99.92

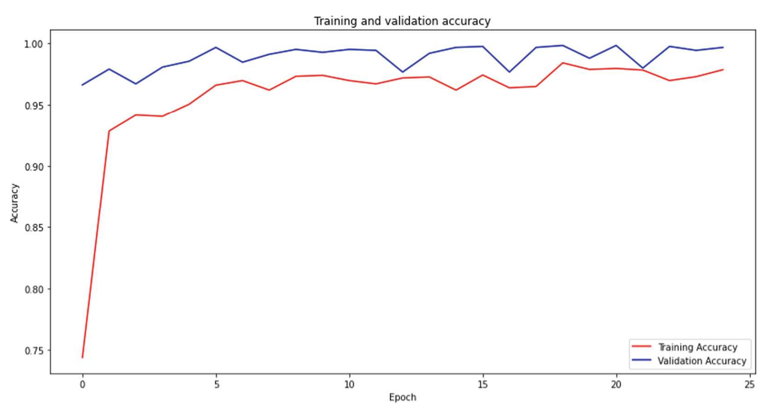

Fig. 4. Accuracy plot for VGG16 transfer learning model's training and validation

From Table 1, we see that the VGG16 training and validation accuracy for the VGG16 model is better. But the scores are close and there is not a significant difference between the scores. The training and validation loss of the two models are shown in Table 2. The losses of both these models are plotted in Figs. 6 and 7.

Unlike the accuracy, in the loss of training and validation, we see a significant difference between the two models (Table 2). The validation loss of the VGG16 model is nearly zero and the training loss is quite lower than the CNN model.

Fig. 5. Accuracy plot for custom CNN model training and validation

Table 2. VGG16 vs CNN training and validation loss

	Modified VGG16	CNN
Training	05.17	09.25
Validation	00.62	01.17

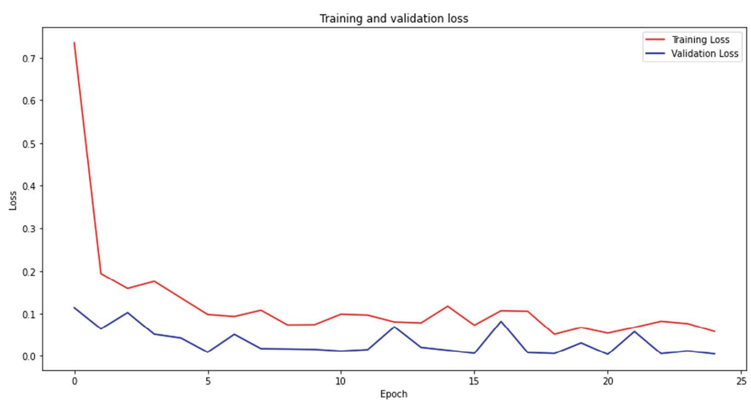

Fig. 6. Plot for VGG16 transfer learning model's training and validation loss

To evaluate the models, we took two approaches. First, we used a test dataset that contained fifty images of each class. The other approach is to test the models with images that are not in the dataset.

For the first approach, we used the test dataset, and the evaluation of the models was done using Keras's built-in model evaluation methods.

Though the VGG16 model performed quite close to the CNN model or even better in terms of training and validation loss, the evaluation accuracy of the model is not very

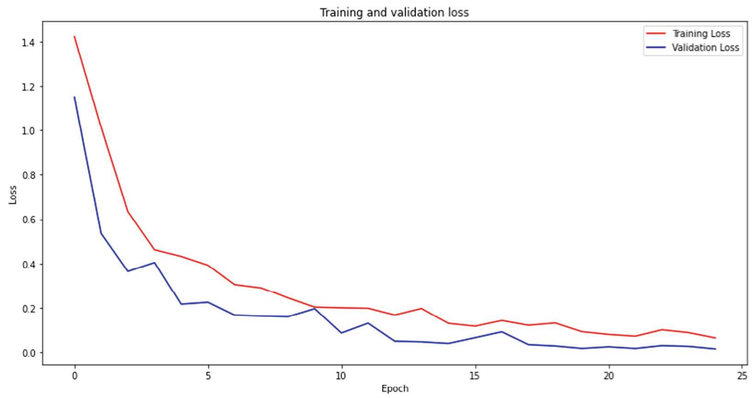

Fig. 7. Plot for custom CNN model's training and validation loss

Table 3. VGG16 vs CNN model evaluation accuracy and loss

	Modified VGG16	CNN
Training	0.75	0.84
Validation	2.87	0.43

impressive. The model has accuracy below 80% while the CNN achieved nearly 85% (as presented in Table 3). The loss in VGG16 was significantly higher than in the CNN model as well.

Now, the other approach to testing can be described in the following steps:

1. Input an image of the target fish class from any other source
2. Predict the image and show the resultant array
3. Sort the resultant array with the name of fish
4. Showing the predicted result.

4.4 Model Testing

The models are tested using various images of the fish classes the models are trained to identify. The result is as below (Table 4).

The VGG16 transfer learning model did not perform better in this test either. Though the model predicted nearly 90% accurately the CNN made from scratch achieved 99.9% accuracy. By the prediction values, the Shrimp class has the second-highest value. However, the VGG16 model false predicted almost 10% that the input image was of a shrimp whereas the CNN model predicted less than 1%.

Table 4. Test image prediction of models

Fish class	VGG16	CNN
Red Mullet	89.273%	99.918%
Shrimp	99.999%	96.351%
Sea Bass	92.348%	98.788%
Gilt Head Bream	91.416%	90.794%
Black Sea Sprat	95.508%	89.472%

5 Conclusion

In this paper, we used transfer learning to classify fishes based on the modified VGG16 CNN model. We also implemented our data augmentation CNN model from scratch with Keras. The data augmentation process performs various image transformations such as zooming, rotation, flipping, etc. to create variation in images to decrease the chances of overfitting. We used five thousand images of five classes for training and validation.

The expected outcome was that the modified VGG16 model would perform significantly better than the CNN model built from scratch. However, that was not the scenario. The transfer learning model performed as well, but not better than the CNN model that was built from the scratch. Regardless of this, both models can identify various fish species with acceptable accuracy. Such models can help in the automatic identification of fishes from aquaculture to household with proper integration or implementation with software applications.

Our findings can help to make deep machine learning models from scratch that can identify fishes more accurately. The paper can prove to be insightful for transfer learning studies. The paper can also help to initiate more research on transfer learning as well as CNN models where researchers can try to increase the classification accuracy of the models. Our findings can also motivate researchers to create more robust and performant deep learning models dedicated to fish classification.

References

AQ1

1. Ogunlana, S.O., Olabode, O., Oluwadare, S.A.A., Iwasokun, G.B.: Fish classification using support vector machine. Afr. J. Comput. ICTs **8**(2), 75–82 (2015)
2. Health Benefits of Fish: Washington State Department of Health. https://doh.wa.gov/community-and-environment/food/fish/health-benefits
3. Fish Pictures and Facts: National Geographic. https://www.nationalgeographic.com/animals/fish
4. Padaruth, S., et al.: SuperFish: a mobile application for fish species recognition using image processing techniques and deep learning. Int. J. Comput. Digit. Syst. **10**, 1–14 (2020)
5. Siraj, F., Salahuddin, M.A., Yusof, S.A.M.: Digital image classification for Malaysian blooming flower. In: 2010 Second International Conference on Computational Intelligence, Modelling and Simulation (2010)

6. Kartika, D.S.Y., Herumurti, D.: Koi fish classification based on HSV color space. In: 2016 International Conference on Information & Communication Technology and Systems (ICTS) (2016)
7. Hridayami, P., Praba, I.K.G.D., Wibawa, K.S.: Fish species recognition using VGG16 deep convolutional neural network. J. Comput. Sci. Eng. **13**, 124–130 (2019)
8. Montalbo, F.J.P., Hernandez, A.A.: Classification of fish species with augmented data using deep convolutional neural network. In: 2019 IEEE 9th International Conference on System Engineering and Technology (ICSET) (2019)
9. Das, A., Roy, S., Bhattacharya, U., Parui, S.K.: Document image classification with intra-domain transfer learning and stacked generalization of deep convolutional neural networks. In: 2018 24th International Conference on Pattern Recognition (ICPR) (2018)
10. Miyazono, T., Saitoh, T.: Fish species recognition based on CNN using annotated image. In: Kim, K.J., Kim, H., Baek, N. (eds.) ICITS 2017. LNEE, vol. 449, pp. 156–163. Springer, Singapore (2018). https://doi.org/10.1007/978-981-10-6451-7_19
11. Iqbal, M.A., Wang, Z., Ali, Z.A., Riaz, S.: Automatic fish species classification using deep convolutional neural networks. Wireless Pers. Commun. **116**(2), 1043–1053 (2021)
12. Desai, C.: Image classification using transfer learning and deep learning. Int. J. Eng. Comput. Sci. **10** (2021). https://doi.org/10.18535/ijecs/v10i9.4622
13. Hussain, M., Bird, J., Faria, D.: A Study on CNN Transfer Learning for Image Classification (2018)
14. Tammina, S.: Transfer learning using VGG-16 with deep convolutional neural network for classifying images. Int. J. Sci. Res. Publ. (IJSRP) **9**(10), 143–150 (2019)