ORNAMENT FISH CLASSIFICATION USING DEEP NEURAL NETWORK

BY

ABIR HASAN NAHID

ID: 161-15-7633

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Aniruddha Rakshit

Department of Computer Science and Engineering

Department of CSE

Daffodil International University

Co-Supervised By

Shah Md Tanvir Siddiquee

Department of Computer Science and Engineering

Department of CSE

Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

JUNE 2021

APPROVAL

This Project/internship titled **"Ornament Fish Classification Using Deep Neural Network''**, submitted by Abir Hasan Nahid, ID No: 161-15-7633 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 02-06-2021.

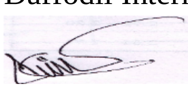
BOARD OF EXAMINERS

________________________ **Chairman**

Dr. Touhid Bhuiyan
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

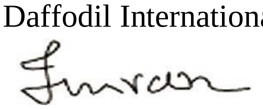
________________________ **Internal Examiner**

Subhenur Latif
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University



______________________ **Internal Examiner**

Md. Abbas Ali Khan
Senior Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University



________________________ **External Examiner**

Shah Md. Imran
Industry Promotion Expert
LICT Project, ICT Division, Bangladesh

DECLARATION

I hereby declare that, this project has been done by me under the supervision of **Aniruddha Rakshit , Department of CSE** Daffodil International University. I also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

____

Mr. Aniruddha Rakshit
Lecturer (Senior Scale)
Department of CSE
Daffodil International University

Co-Supervised by:

____

Mr. Shah Md Tanvir Siddiquee
Assistant Professor
Department of CSE
Daffodil International University

Submitted by:

____

(Abir Hasan Nahid)
ID: 161-15-7633
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First I express my heartiest thanks and gratefulness to almighty God for His divine blessing makes me possible to complete the final year project/internship successfully.

I really grateful and wish my profound my indebtedness to **Aniruddha Rakshit**, **Sr.Lecturer**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of my supervisor in the field of Deep Learning to carry out this project. His endless patience ,scholarly guidance ,continual encouragement , constant and energetic supervision, constructive criticism , valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project.

I would like to express my heartiest gratitude to all mighty Allah, and Head**,** Department of CSE, for his kind help to finish my project and also to other faculty member and the staff of CSE department of Daffodil International University.

I would like to thank my entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, I must acknowledge with due respect the constant support and patients of my parents.

ABSTRACT

Ornamental fish are the most important fishing commodity in the world in terms of unit weight, with a global export volume of about 350 million dollars. Ornamental fish are grown using the same simple technology and nutrients as food fish, but ornamental fish farming has not historically been considered an aquaculture practice. This is due in part to the fact that literature on different facets of ornamental fish culture is often conducted in books and specialized magazines rather than peer-reviewed publications, or it is held proprietary. Technology has the potential to make a significant contribution to all facets of our lives. Deep learning algorithms that use a particular kind of neural network called a convolutional neural network (CNN) to make sense of images are at the heart of today's computer vision technologies. In deep learning, I will use the convolutional neural network (CNN) to achieve state-of-the-art precision in a variety of classification problems, such as image data, CIFAR-100, CIFAR-10, and MINIST data sets. In this paper, I propose a novel system that uses Convolutional neural networks to classify different types of ornament fish detections, automatic self-ruling decision making, and predictive models (CNN). While there has been a number of studies on fish picture detections in image classification problems in the past, our associated tropic ornament fish type detection issue has just a few works on various data sets and different models with low accuracy. I retrained the final layer of the CNN architecture, VGG16, Inception V3 for classification strategy, for solid architecture. Predicting between five groups (goldfish, Arowana fish, betta fish, angelfish, rainbow shark). I suggested a 95% average accuracy that can be used for a variety of uses, such as purchasing fish, classification and assisting in the management of a large aquarium.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

1.1 Introduction

Computer vision and artificial intelligence have been extensively used for human behavior recognition in recent years, such as facial recognition, understanding emotion, fish expression detection, and so on. Technology has been used in a variety of ways. Using machine vision software, I will use fish image recognition technologies to classify fish. People in every country now enjoy keeping ornamental fish aquariums and devoting time to maintaining the aquarium and its inhabitants. Nowadays, aquarium fishing is becoming more common in our world, so I can create an automated decision-making application to use in fish image identification to aid aquarium management and correctly locate fish. I can use deep learning to retrieve features from an image by using a convolutional neural network (CNN) that reads every pixel of the image. There are several different kinds of fish, but I only deal with five popular ones: Goldfish, Arowana fish, Betta fish, Angelfish, and Rainbow shark. My main focus is to create a transfer learning model that can identify various types of fish in photos and can be used in computer vision applications. To do so, I must first train my algorithm with various types of models; however, I created a new CNN model to train my images for greater accuracy. And also train VGG16, InceptionV3 models for comparison with my CNN model for more trustworthy. There are several parts to complete my task, for easy to understand I can divide my task into different following sections, such as in section 3 described my proposed methodology that's included in an implementation of my model, data collection, data augmentation, data preprocessing, define test set for evaluating my models and train data set to train my model. In section 4 are described as performance evaluation, result discussion, comparison. And future work and conclusions and future works are mentions in section 5.

1.2 Motivation

Technology is an integral component of every human being's because with technology we can make easier our daily life. Now we can use artificial intelligence that able to make a decision like humans. When we focus on an object or something then in a short time we can able to make a clear sense of that object. Our brain are very strong and faster to make a clear sense, but it is not easy for us. Every time we processing a minimum of 60 images with high-resolution pixels to recognize that. When an object's light go through our retina with neurons to our brain then we can see and understand about that object but take a little bit of time to give a result. To understand something we have to train our brain, like that if we want to make a machine that can see the object and identify it successfully. For that, we have to teach a machine to classify objects near humans. At that idea, I can build a model that can classify an image with a relation of the fishes.

1.3 Rationale of the Study

Using artificial intelligence computers are becoming more and more human Because there is no department we don't use a computer. We can use artificial intelligence anywhere like as the internet, playground, home, office, factory, etc. For helping classification system we can use it in a different way and different places. By considering that's facts I can make a model that successfully helps a human to find aquarium fishes correctly. My proposed model makes a vital role in successfully recognized ornament fishes efficiently. For making a model we can use various machine learning models like deep learning using a convolutional neural network.

1.4 Research Questions

For the first time, I'm not sure what my first job. since determining my best direction is very difficult for me. At the moment, I was perplexed and had some doubts.

- What programming language is best for solving my issues?
- Which picture forms perform exceptionally well.
- What are the benefits to the people?
- Is a 90 percent accurate classification rate possible?
- How easy is it for me to do that?

1.5 Expected Outcome

I want to build a model that will effectively direct decision-making. The model reliably detects ornament fish images and can forecast for decision-making assistance. For the best results, I want to create a model using common most recent machine learning techniques in my project. My model accurately recognizes various types of fish and provides guidance. My model is equally at home inside and out. It is capable of correctly predicting 90% of all inputs. my model will be lightweight because it takes a little bit of time to give outputs for every input. My model is a read-only image from a camera, video, text file, or manually entered the information. However, the model's performance can be interpreted in a variety of ways. I used Convolutional Neural Network Deep Learning to build my concept. Since the convolutional neural network performs well in image data-set classification. There are a few key objectives for the project's result.

- Images of ornamental fishes have been classified.
- A small amount of time to produce results.
- It assists people in locating accurate fish by providing information.
- It has the ability to work both indoors and outdoors.

1.6 Layout of the Report

In this chapter one, I already discourse about the introduction of my project, objective, motivation, research questions, and last was expected the outcome of my project

Chapter 2

2.1 Introduction

In this segment, I discuss about my project's related works, background details, and the nature of the problems and challenges. Since I can find relevant information from my related works or literature reviews, the discussion is helpful for my project in solving some problems. In this section, I will determine the complexity of the issues as well as the obstacles that must be overcome to improve accuracy.

2.2 Related Works

Before the widespread works on image classifications with different data sets or different algorithms. But till now, there are few image-based ornament fish detection worked. A still fish image I can classify and represent in many different ways. The CNN's ability to extract details for objects of interest based on their color, texture, and shape is enhanced by training on datasets with significant variations in context and objects in the images. As a result, any visual pattern could be easily captured and learned by the appropriate network. The network generalization capacity grows as the number of examples for a particular object grows. This generalization capability allows the qualified network to identify knowledge that isn't used for training. Pre-trained deep CNNs include Inceptionv3, Vgg-16, and Vgg-19. Each deep CNN's expertise can be used to train and evaluate on a larger number of datasets. Deep CNN knowledge transfer is one of the key benefits that improve the usability and accuracy of deep neural networks.

2.3 Research Summary

Machine learning research began with simple algorithms, but as the complexity of machine learning algorithms increased, it became possible to solve increasingly complex problems.

As a result, I can conclude that machine learning is becoming increasingly important in all aspects of our lives. Machine learning methods have been used in image-based classification for a few years now. I can teach machines to understand images using convolutional neural networks. There are several works in this picture classification section that are done in various ways. From the history of machine learning and related works, I gain a better understanding. I may now state that there are some works on aquarium fish identification, but I created a new model that can more effectively and reliably classify ornament fishes.

2.4 Scope of the problem

There are so many sections for the problem in and phase of my mission. I can't identify problems once I start a new job, but I can understand that during the study. After gathering data, I must manipulate it in some way before training. The most difficult tasks for working are programming syntax and model selection.

2.5 Challenges

There are some obstacles to overcome to complete my project. To train my model, I must first collect data from various platforms. Before using the CNN model, I need to prepare details. The biggest challenge was creating a new model with several layers. For me, defining a training range, avoiding Overfitting, selecting a number of the epoch, and batch size are more difficult. My mission is extremely difficult, but it is possible to achieve it successfully.

Chapter 3

Research Methodology

3.1 Introduction

In the third chapter, I'll go over my methodology in detail. I use a convolutional neural network for image classification in this study because it is the most efficient and widely used deep learning algorithm. I'm going to use my own data set and Convolutional Neural Network model that I developed (CNN). I go through the training and test sets, input, and output, per convolutional layer, various technical issues, results, and so on. I'll provide a good example of the cleaning principle.

3.2 Research Subject and Instrumentation

The research subject makes a clear concept of my research area. In this section, I implement and design my model, collect perfect data, prepare them and train my model, performance discussion, and then apply my model to work. To complete my task I used windows platform. For implementing my task I use python programing language and many other packages like Keras, TensorFlow, numpy, cv2, seaborn, matplotlib, etc. I use python's another package mini-conda to complete all programming tasks on jupyter notebook. I choose Python because of it free, simple syntax, foster fast testing for complex algorithms readability for machine learning applications.

3.3 Data Collection Procedure

To train the proposed networks in this analysis, a new dataset was developed. Such photos were gathered from a variety of sources, including websites, Facebook, Instagram, Reddit, and other social media networks. The resolution and format of all PNG images are not the same. There are over 4440 photos in five groups, including goldfish, Arowana fish, betta fish, angelfish, and rainbow shark. This dataset was compiled by hand from publicly accessible information on the internet. There are over 700 images in each class. There are a total of 4440 images in this dataset, with 3552 images for training and 20% of 888 for research.

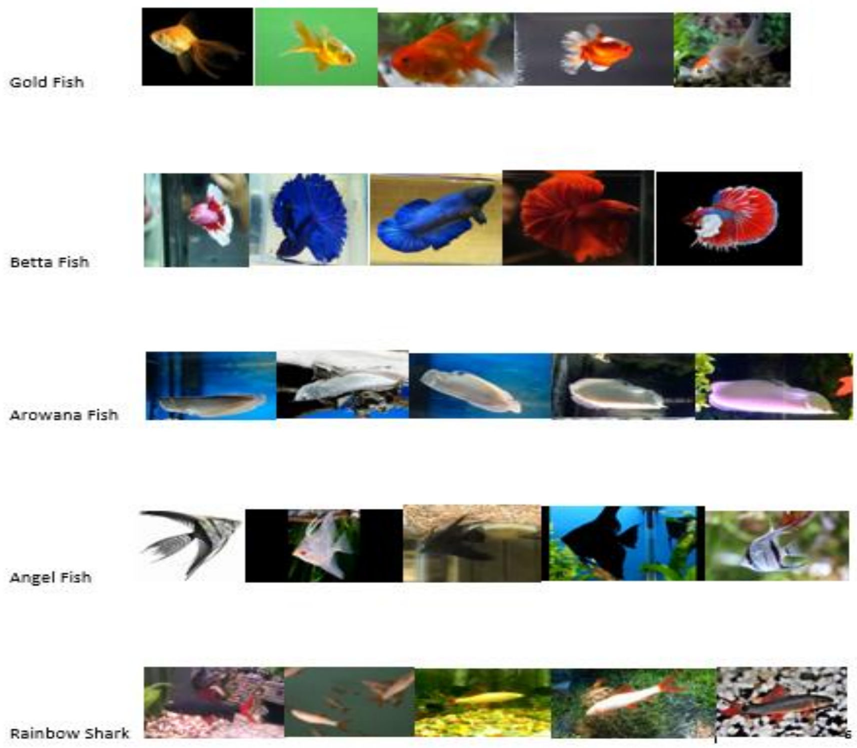


Fig 3.3: A small part of the data set

3.4 Data Processing

My first task is to gather data and process it. When I collect my data collection, I find that not all of the images have the same resolution, so I have to prepare them for work. Data processing is crucial for training the model and improving accuracy because it lowers computational costs, reduces overfitting, and so on.

3.5 Data Augmentation

Overfitting will occur after a few training sets. To prevent overfitting, the number of new dataset samples was increased using basic types of image increase, which I used to train my convolutional neural network model(CNN). Data augmentation is critical because it improves model efficiency while lowering classification loss. I compiled total data using three separate methods, each of which is depicted in the diagram below:

- Flip horizontally about Y-axis
- rotate left -30 degree
- to rotate right +30 degree

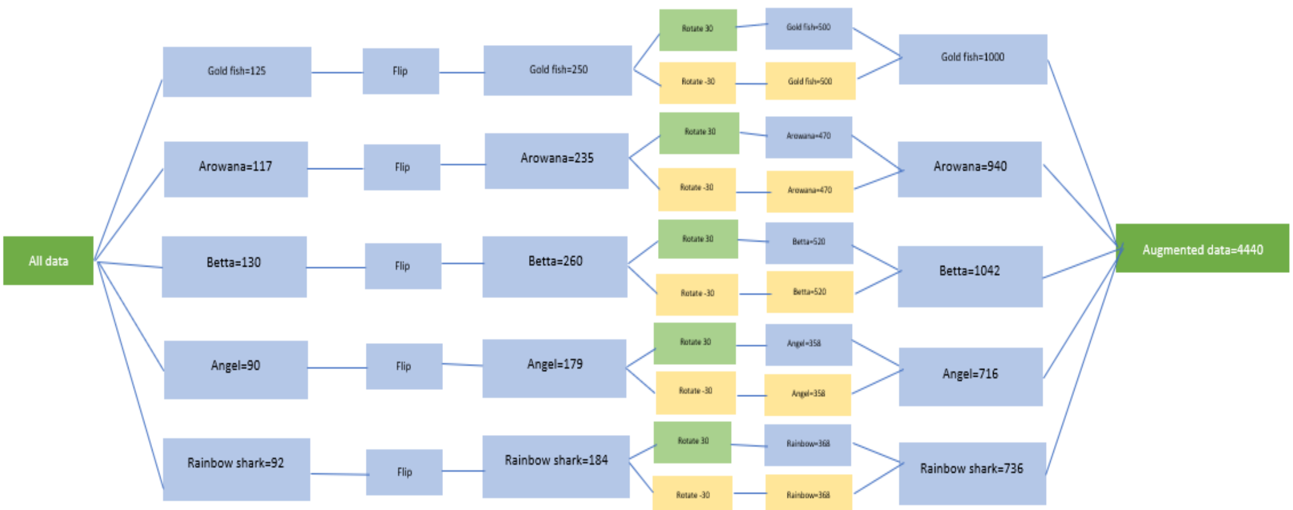
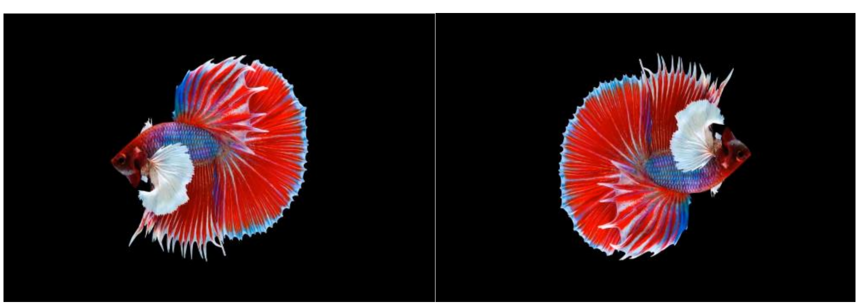


Fig 3.5 Data Augmentation



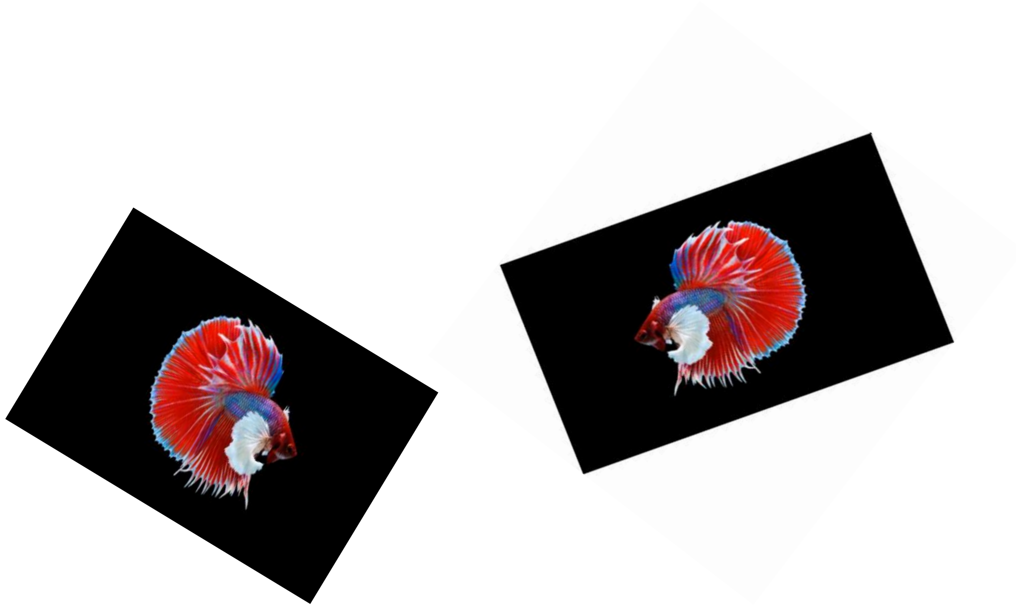


Fig3.5:Sample of data

3.5.1 Data Preparation

After augmentation for avoiding costs more computation resources and a chance of overfitting, I can reduce the input dimension fixed into 150 X 150 pixels of images. Before passing images to CNN I used RGB color that easily helps to detect features by CNN that ensure that to get better accuracy. For normalizing I can reduce RGB values dividing by 255 and get the range of [−0.5, 0.5].

3.6 Proposed Methodology

Convolutional neural network (CNN) is a specific type of artificial neural network architecture for deep learning that uses perceptron, a grating machine learning unit algorithm, for supervised learning, to analyze different types of data. CNN operations generally work depends on inputs for extracting pattern recognition and it works well with data that has a spatial relationship CNN also has a learnable parameter like neural network i.e., weights, biases, etc. Some of these layers are convolutional, using a mathematical operation and model to pass on results to successive layers.

3.6.1 Proposed Methodology Convolutional Layer

Convolutional neural network (CNN) is a specific type of artificial neural network architecture for deep learning that uses perceptron, a grating machine learning unit algorithm, for supervised learning, to analyze different types of data. CNN operations generally work depends on inputs for extracting pattern recognition and it works well with data that has a spatial relationship CNN also has a learnable parameter like neural network i.e., weights, biases, etc. [5]. Some of these layers are convolutional, using a mathematical operation and model to pass on results to successive layers.

3.6.2 Convolutional layer

A convolutional neural network has a basic structure of an input layer, an output layer, and various hidden layers. Each input is convoluted with various types of the filter during the forward propagation (or kernel). I can apply in Image data, Classification prediction problems, Regression prediction problems, face recognition, object detection, segmentation, etc. There are several layers to classify an image such as CNN's first layer is the input layer that read-only image pixel to pixel, that image may be RGB of grayscale category. Then the second step uses filter to extract the feature of the input image in various ways. Different types of *pooling* layers use to reduce input images shape to reduce parameters. The result of the convolution shows momentum which affects the classification. These samples are called features. To build a convolutional layer, some model hyper-parameters have to be configured: length of filters, number of filter, stride, and padding.

- Length of filters: Kernel acts as a filter, it's working for extracting specific features or patterns identifications in the input data, which make increase the efficiency of the classifications. In CNN's, filters are not defined. The value of every filter is learned throughout the training process. By humans able to learn the values of different filters, CNNs can find more meaning from input images by filtering but human-designed filters provably not be able to find specific features.

  Filter could be many different sizes like as 3X3 filter, which is small but probably easy to read input image pixels.

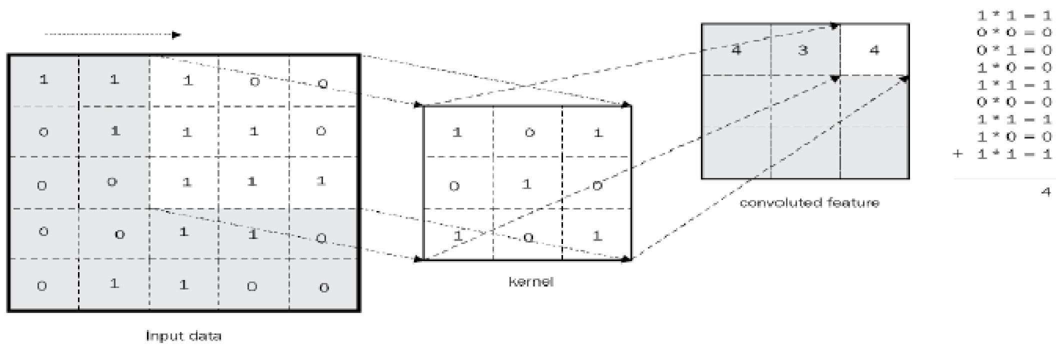


Fig 3.6: Kernel

- Stride: Stride defines the number of rows and columns that shifts pixels over the input matrix. Stride reduces the output dimension if the input matrix. If stride is 2 then move the filters to 2 pixels of the input matrix. Stride number always an integer and not a fraction, by default stride is 1.

- Padding: After the convolution layer reduces the dimensions of the output matrix, but using padding I can maintain the dimension of output as an input matrix. There are two kinds of padding: the same padding and valid padding. Valid padding means "no-padding", it reduces the dimension of the output matrix. The same padding means the output matrix is the same dimension as the input matrix. In the same padding adding an extra block and assign zero to the input matrix symmetrically for the same dimension.

Use the activation function (denoted σ) to identify those features that are relevant for classification after each convolutional operation.

3.6.3 Rectified Linear Units (ReLU):

Rectified Linear Units (ReLU): ReLU is the most widely used activation function while designing networks today. ReLU function is nonlinear and allows for backpropagation. The constant gradient of ReLUs results in faster learning because it does not activate all the neurons at the same time like as if the input is negative it will convert into zero and that neuron does not get activated. So few neurons are active at a time, not all neurons, at this reason ReLU much easier, faster and make more efficient. More biological inspired to train.

3.6.4 Pooling layer:

Pooling layer is a non-linear layer that divides the dimension of the input and reduces the number of parameters, controlling overfitting and most relevant information is preserved.

I can define the size of the PoolingLayer that can remove unnecessary features and keep the necessary features. There are three kinds of pooling layer *MaxPooling*, *AveragePooling*, and *MinPooling*. In deep learning, three pooling functions.

- MaxPooling: It picks only the maximum value contained in the pooling window. In CNN architecture MaxPooing is mostly used for recognizing relevant features because MaxPooling gives a better result from AveragePooling and MinPooling layer.
- AveragePooling: It picks only Average value contained in the pooling window.
- MinPooling: It picks only the Minimum value contained in the pooling window.

3.6.5 Flatten Layer:

Flatten Layer In between the convolutional layer and fully connected layer, there's a 'Flatten' layer. It's converting all the resultant 2-dimensional arrays into a 1D feature vector, this operation is called flattening. This flattening structure makes a single long continuous linear vector to be used by the dense layer for the final classification layer.

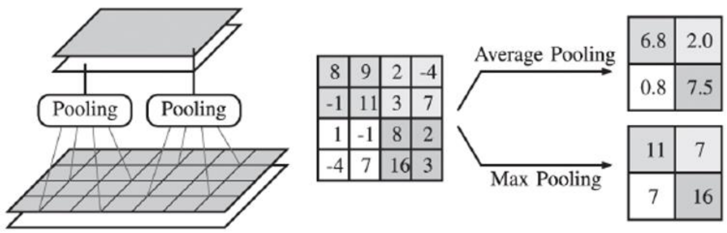


Fig 3.6.5: Flatten Layer

3.6.6 Fully connected layer:

Fully connected layer is the last phase for a CNN network, it represents the feature vector for the input. In FC layer involves weights, biases, and neurons. It connects neurons in one layer to neurons in another layer. FC layers recombine each neuron to efficiently and accurately classify each input. It is used to classify images between different categories by training. FC layers can be contrasted with Multilayer Perceptron (MLP) where each neuron has complete links with all previous layer activations.

3.6.7 Dropout layer: The dropout layer forces a neural network to learn more strong features that are useful in conjunction with many different random subsets of the other neurons. This layer layers improve over-fitting, reduce dependence and complexity on the training set.

3.6.8 Softmax layer:

Softmax layer is the last layer or output layer in neural network functions and its use for determining the probability of multiple classes. This function calculate the probabilities of each target class and return the values to determining the target class for the given inputs.

3.7 Test Set

This dataset includes 5 different classes and contains an average of 110 core images for each classes. After augmentation total 4440 images for all classes. In this part, I make a test set to evaluate the classification performances of my CNN model. After data preprocessing I have split into two different portion test-set and train-set. First, train-set uses to train my model. When I successfully trained my model then I can evaluate my model using test-set. I select test-set and train-set using random state=42 to get more valid accuracy and performed well on the unseen test set. 80% of the 4440 images are used to create the training set and the remaining 20% is used for the test set. So in train-set are contains a total of 3560 images and test-set 880 images.

3.8 Training the Model

After generating data preprocessing and defining train set, test set then I ready to train my model with training data sets that consist of 3560 images. For increasing accuracy and decreasing loss as possible, I update my model many times and change the optimizer, learning rate, loss function. I train my model using Adam optimizer to reduce loss function as possible and applied on a 80% training set and 20% validation set. I use 128 batch size for less memory and faster for training my model. As the process continues, then I can see that in 70 to 80 number of epochs training accuracy and validation accuracy are not increased significantly. At that stage validation accuracy reached 95.27%, training accuracy 9681%, and validation loss 0.1422%. Then my model is ready to predict unseen data for final test evolution.

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 150, 150, 32)	2432
max_pooling2d (MaxPooling2D)	(None, 75, 75, 32)	0
conv2d_1 (Conv2D)	(None, 75, 75, 64)	18496
max_pooling2d_1 (MaxPooling2	(None, 37, 37, 64)	0
conv2d_2 (Conv2D)	(None, 37, 37, 96)	55392
max_pooling2d_2 (MaxPooling2	(None, 18, 18, 96)	0
conv2d_3 (Conv2D)	(None, 18, 18, 96)	83040
max_pooling2d_3 (MaxPooling2	(None, 9, 9, 96)	0
flatten (Flatten)	(None, 7776)	0
dense (Dense)	(None, 512)	3981824
activation (Activation)	(None, 512)	0
dense_1 (Dense)	(None, 6)	3078

Total params: 4,144,262
Trainable params: 4,144,262
Non-trainable params: 0

Fig 3.8: Training the Model

3.9 Implementation Requirements

After fully describing my CNN methodology and completely trained my mode a requisite list was created and those requirements are must essentially need for image-based classifications.

Hardware/Software Requirements

- Operating System (Windows 7 or above)
- Hard Disk (minimum 500 GB)
- Ram (Minimum 4 GB)
- GPU(Recommended)

Developing Tools

- Python Environment
- jupyter notebook (Anaconda3, Mini-Conda3)

CHAPTER 4

EXPERIMENTAL RESULTS AND DISCUSSION

4.1 Introduction

In chapter 4, I discuss the performance evaluation of my model, number of parameters, accuracy level. I compare my model with InceptionV3 and VGG16. For easily undergrad, I use graphs, pictures and confusion matrix.

4.2 Performance Evaluation

In order to find the best practices, I applied the proposed CNN architecture to the datasets described above and achieves significantly better results in average 95 per-class accuracy. In this picture, we can see the loss is going downwards, and at this time validation accuracy are increasing, so at that stage, I can say that the training model learns perfectly from training data set.

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 (%)	Number of epochs
Inception V3	97	97	98	97	60
VGG16	95	96	96	96	80
CNN	99	99	99	99	80

Table 4.2: Performance Evaluation

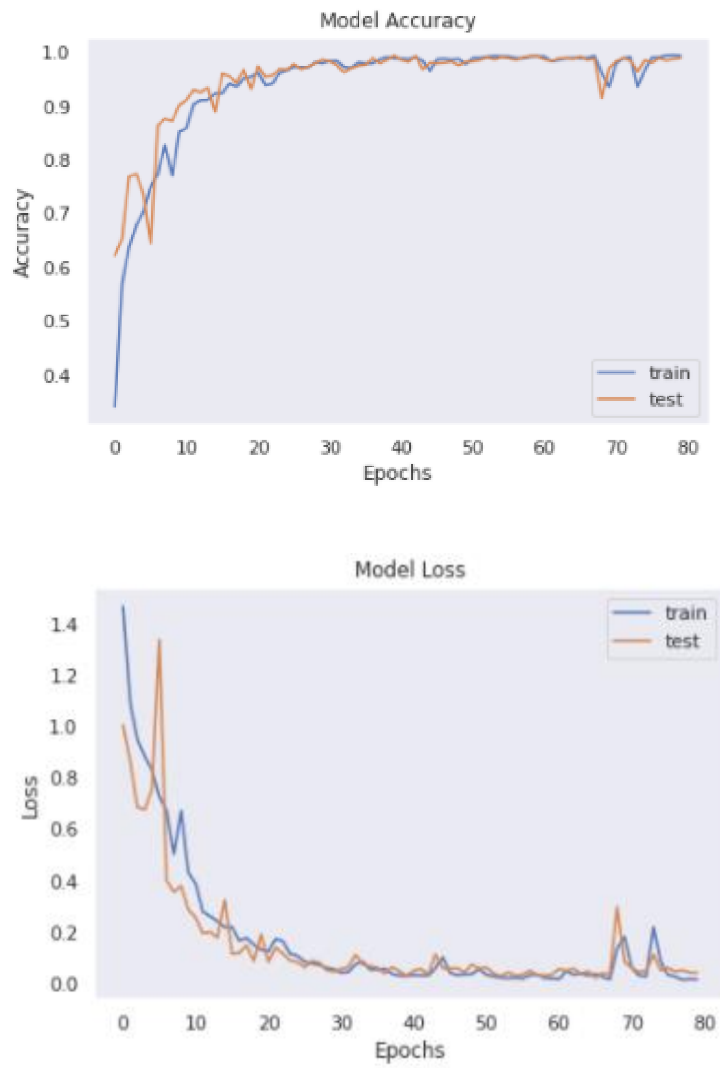


Fig4.2.1:Epochs

4.3 Number of Parameters

In my convolutional network model decorated with some of different layers, input size, number of filters, activation shape, etc. that generate weights, biases and that makes total 4,144,262 numbers of the parameter. First input image shape is 150X150 using RGB color to read-only image without generating parameters. There are used 4 numbers of Conv2D layers and MaxPool layers one Flatten layer, 2 dense layers with one output layer or softmax layer. The pool size(2,2), strides(2,2) are same but number of filters(32,64,96,96) and kernel size (5,5), (3,3) are different in different layers.

Number 1: In the input layer using for reading image pixel to pixel. For Input_Size that's activation shape are (150X150X3) for that activation size are 67,500 because input image wide 150, hight 150, and 3 for RGB color. In the input, the layer is no parameters.

Number 2: Conv2D 1 is the first layer of the convolutional network. It mainly works to extract features from an image with filters. In this model 32 number filter used and for that activation shape is (150X150X32) and activation size is 720,000. In Conv2D 1 generate (((5X5X3)+1)X32)=2,432 number parameters. Here (5X5) is kernel size and for RGB color 3.

Number 3: MaxPooling layer use to reduce the number of parameters, avoid the chance of overfitting. Here pooling size f=(2X2), stried s =2, Image wide w = 150, hight h = 150. So activation shape formula are (w-f+1)/s. so activation shape is ((150-2+1)/2) =74.5 =75. Pooling layer educe the image dimension 150 to 75 for that activation size is (75X75X32)= 180000. Here 32 is filter size. The pooling layer doesn't generate parameters.

Number 4: Conv2D 2 is the second layer of my model. It complete the same work for activation size (75X75X64) =360000 activation shape, and with 64 is number of filters make (((3X3X32)+1)X64)=2,432 18,496 number of parameters. Here (3X3) are kernel size, and 64 is filter size.

Number 4: In section 4 contain MaxPool 2. It is the second layer of MaxPool 2 for this model. At that same way, pooling layer reduce the number of image shape 75 to 37 and make activation size 87616 without generating parameters.

Number 10: After the convolutional layer and pooling layer are flattened layer. It's converting all the resultant 2-dimensional arrays into a 1D feature vector. Activation shape (77761X1) = 77761number of activation size.

Number 11: In this section is a dense layer or fully connected layer. In dace, layer has activation shape (512) but no activation size. It generates only parameters ((1+77761)X512) = 3,981,824.

Number 12: Softmax layer is the last layer of this model. I can say that it is the output layer of my model. The softmax layer generates parameters with a final shape of output. Here (6X(512+1)) = 3078 number of parameters.

After adding a total number of parameters is 4,144,262. CNN's Conv2D and MaxPool layers generate activation size using activation shape. MaxPool layer uses for reducing dimension size, and there are no parameters. Total parameters generate only Conv2D layers. For stable generalization, I don't increase CNN layers.

Layer		Number of Filter	Activation Shape	Activation Size	Parameters
Number	Operation				
1	Input Size	-	150,150,3	67,500	-
2	Conv2D 1	32	150,150,32	720,000	2,432
3	MaxPool 1	-	75,75,32	180,000	-
4	Conv2D 2	64	75,75,64	360,000	18,496
5	MaxPool 2	-	37,37,64	87,616	-
6	Conv2D 3	96	37,37,96	131,424	55,392
7	MaxPool3	-	18,18,96	31,104	-
8	Conv2D 4	96	18,18,96	31,104	83,040
9	MaxPool4	-	9,9,96	7,776	-
10	Flatten	-	7776,1	7,776	-
11	Dense 1	-	512	-	3,981,824
12	Softmax	-	6	-	3078
Total Parameters					4,144,262

Table 4.3: Number of Parameters

4.4 Result Discussion

We know that the classifier's performance was established on a test set from the training, validation and testing accuracy. My CNN model give a high accuracy of precision, recall and every weighted average up to 98. Total test dataset images are 1110, and after classification, only 42 images are false predictions, another way 1043 is a correct prediction. The final test of my model gives 99% accuracy, so my model gives a better test accuracy for unseen data. To make a clear assume we can observer the confusion matrix.

Class	Precision (%)	Recall (%)	F1 (%)
Gold Fishes	96	100	98
Betta Fish	99	100	99
Arowana	99	99	99
Angel	100	98	99
Rainbow Shark	100	99	99

Table 4.4: CNN Classification Report

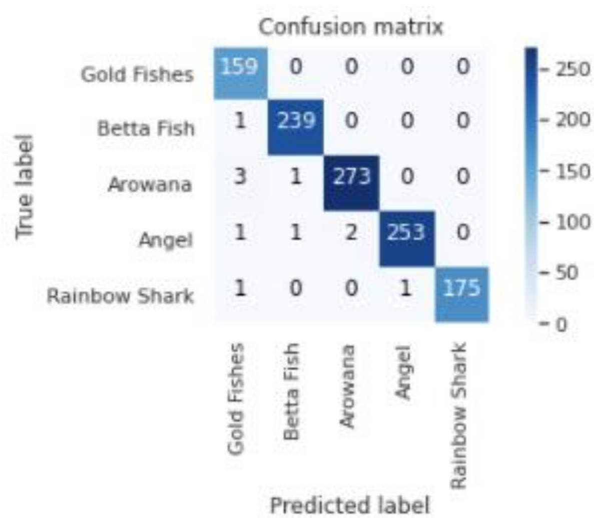
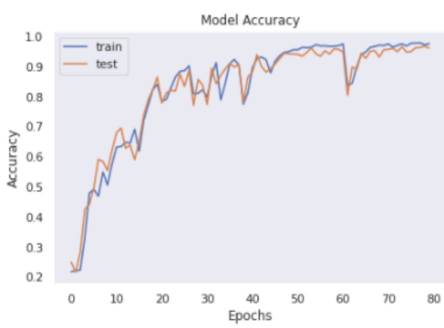


Fig 4.4: Predicted label

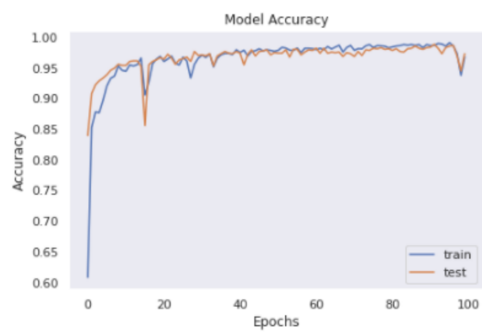
4.5 Comparison

In this section, I compared different models with my CNN models such as Inception-v3, VGG16 with their test accuracies and used the same number of epochs and batch-size are used for best comparison. For every model to train used 25% of test-set and 75% train-set are the same but validation accuracy and testing accuracy are different by a different model that is summarized in table 2[9]. From the table, we observed that VGG16 gives 96% accuracy with a little bit noisy, and validation accuracy and train accuracy rate often similar. InceptionV3 gives low validation accuracy for these datasets. But my CNN model give the highest validation accuracy for a good train accuracy with little bit noisy. Finally, we observed that under the same conditions, every model's accuracy is satisfied, but among them, My CNN performs perfectly with a validation accuracy of 99%.

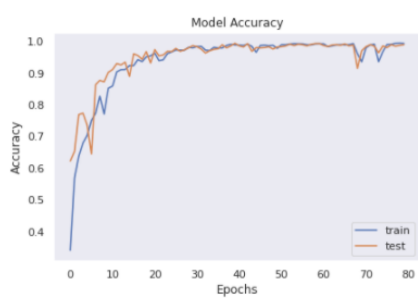
I proposed a new CNN architecture and that achieved the high testing accuracy from another model.



VGG16



Inception V3



CNN

Fig 4.5: Comparison

CHAPTER 5

CONCLUSION AND FUTURE WORKS

5.1 Future Work

My proposed model CNN shows better accuracy for classification against inceptionV3, VGG16, and for different fish image detection. But in the future, there are several ways to update my model for transfer learning. I'll apply another various model such as *AlexNet*, *ResNet* to increasing accuracy, efficient training and feature extraction for all models and other hand GPU are the most important for training. But I suggest a strong approach apply ensemble method for the best accuracy and I shall test other advanced classification ideas, such as transfer learning.

5.2 Conclusions

In this paper, I build up a model using CNN architecture and its competitive classification accuracy performance up to 99%. And I described the number of filters, Activation shape, Activation size, total parameters and number of convolutional blocks of my model. My model is simple but performed much faster with high accuracy from another complex model. Finally, I demonstrate that the proposed method has high accuracy from the experiment. At last, I can say that CNN can be used to improve the object classification capacities of different areas in a new and innovative way.

References

[1] Fouad, M.M.M., Zawbaa, H.M., El-Bendary, N., Hassanien, A.E.: Automatic Nile Tilapia fish classification approach using machine learning techniques. In: 13th International Conference on Hybrid Intelligent Systems, HIS 2013, pp. 173–178. IEEE(2013)

[2] Fouad, M.M., Zawbaa, H.M., Gaber, T., Snasel, V., Hassanien, A.E.: A fish detection approach based on BAT algorithm. In: The 1st International Conference on Advanced Intelligent System and Informatics, AISI 2015, pp. 273–283. Springer, Beni Suef(2016)

[3] Dominguez, A.: A history of the convolution operation [Retrospectroscope]. IEEE Pulse 6, 38–49(2015).

[4] Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE 86, 2278–2324(1998).

[5] Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems, pp. 1097–1105. Curran Associates Inc.(2012)

[6] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet large scale visual recognition challenge. Int. J. Comput. Vis. 115, 211–252(2015).

[7] Jiang, H., Learned-Miller, E.: Face detection with the faster R-CNN. In: 2017 12th IEEE International Conference on Automatic Face and Gesture Recognition, FG 2017, pp. 650–657. IEEE(2017)

[8] Strachan, N., Kell, L.: A potential method for the differentiation between haddock fish stocks by computer vision using canonical discriminant analysis. ICES J. Mar. Sci. 52, 145–149(1995).

[9] Rova, A., Rova, A., Mori, G., Dill, L.M.: One fish, two fish, butterfish, trumpeter: recognizing fish in underwater video. In: IAPR Conference on Machine Vision Applications, Tokyo, Japan, pp. 404–407(2007)

[10] Spampinato, C., Giordano, D., Di Salvo, R., Chen-Burger, Y.-H.J., Fisher, R.B., Nadarajan,G.: Automatic fish classification for underwater species behavior understanding. In:Proceedings of the First ACM International Workshop on Analysis and Retrieval of Tracked Events and Motion in Imagery Streams, ARTEMIS 2010, p. 45. ACM Press, New York(2010)

[11] Khalifa, N.E.M., Taha, M.H.N., Hassanien, A.E., Selim, I.M.: Deep galaxy: classification of galaxies based on deep convolutional neural networks(2017).

[12] Sainath, T.N., Kingsbury, B., Mohamed, A., Ramabhadran, B.: Learning filter banks within a deep neural network framework. In: 2013 IEEE Workshop on Automatic Speech Recognition and Understanding, pp. 297–302. IEEE(2013)

[13] Khalifa, N.E., Taha, M.H., Hassanien, A.E., Selim, I.: Deep Galaxy V2: robust deep convolutional neural networks for galaxy morphology classifications. In: 2018 IEEE International Conference on Computing Sciences and Engineering, ICCSE, pp. 122–127. IEEE(2018)

[14] Bui, H.M., Lech, M., Cheng, E., Neville, K., Burnett, I.S.: Object recognition using deep convolutional features transformed by a recursive network structure. IEEE Access 4, 10059–10066(2017).

[15] Scott, G.J., England, M.R., Starms, W.A., Marcum, R.A., Davis, C.H.: Training deep convolutional neural networks for land-cover classification of high-resolution imagery. IEEE Geosci. Remote Sens. Lett. 14, 549–553(2017).

EXNET: ORNAMENT FISH CLASSIFICATION USING DEEP NEURAL NETWORK

ORIGINALITY REPORT

14%	%	%	14%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Coventry University Student Paper	1%
2	Submitted to Netaji Subhas Institute of Technology Student Paper	1%
3	Submitted to Higher Education Commission Pakistan Student Paper	1%
4	Submitted to Daffodil International University Student Paper	1%
5	Submitted to Study Group Australia Student Paper	1%
6	Submitted to Cebu Institute of Technology Student Paper	1%
7	Submitted to Vignan's Foundation for Science, Technology and Research University Student Paper	1%
8	Submitted to Leeds Beckett University Student Paper	1%

9	Submitted to University of Oxford Student Paper	1 %
10	Submitted to University of Hong Kong Student Paper	1 %
11	Submitted to American Intercontinental University Online Student Paper	1 %
12	Submitted to Bayard Rustin High School Student Paper	<1 %
13	Submitted to Seoul National University Student Paper	<1 %
14	Submitted to University of Central Oklahoma Student Paper	<1 %
15	Submitted to Misr International University Student Paper	<1 %
16	Submitted to Technische Universiteit Delft Student Paper	<1 %
17	Submitted to College of Engineering, Pune Student Paper	<1 %
18	Submitted to Georgia Institute of Technology Main Campus Student Paper	<1 %
19	Submitted to Feng Chia University Student Paper	<1 %

20	Submitted to AUT University Student Paper	<1 %
21	Submitted to Liverpool John Moores University Student Paper	<1 %

Exclude quotes Off

Exclude bibliography Off

Exclude matches Off