NEIL GOGTE INSTITUTE OF TECHNOLOGY

A Unit of Keshav Memorial Technical Education (KMTES)

Approved by AICTE, New Delhi & Affiliated to Osmania University, Hyderabad

A

SUMMER INTERNSHIP REPORT

ON

IMAGE BASED BIRD SPECIES IDENTIFICATION

For the award of Degree of

BACHELOR OF ENGINEERING

IN

CSE (AI ML)

Submitted By

SRIMANTH M. **245320748045**

Under the guidance of

Mrs. M. DEEPIKA

Department of CSE(AIML)

NEIL GOGTE INSTITUTE OF TECHNOLOGY

Kachavanisingaram Village, Hyderabad, Telangana 500058.

JANUARY 2023

NEIL GOGTE INSTITUTE OF TECHNOLOGY

A Unit of Keshav Memorial Technical Education (KMTES)

Approved by AICTE, New Delhi & Affiliated to Osmania University, Hyderabad

CERTIFICATE

This is to certify that the Mini project work entitled "**IMAGE BASED BIRD SPECIES IDENTIFICATION"** *is a bonafide work carried out by* **SRIMANTH M. (245320748045)** of III-year V semester **Bachelor of Engineering** *in CSE(***AIML)** *by Osmania University, Hyderabad during the academic year* **2022-2023** *is a record of bonafide work carried out by them. The results embodied in this report have not been submitted to any other University or Institution for the award of any degree*

Internal Guide	**Head of Department**
Mrs. M. Deepika	Dr. K. Madhuri
Assistant Professor	Associate Professor

External

NEIL GOGTE INSTITUTE OF TECHNOLOGY
A Unit of Keshav Memorial Technical Education (KMTES)
Approved by AICTE, New Delhi & Affiliated to Osmania University, Hyderabad

DECLARATION

I hereby declare that the Mini Project Report entitled, "**IMAGE BASED BIRD SPECIES IDENTIFICATION**" submitted for the B.E degree is entirely my work and all ideas and references have been duly acknowledged. It does not contain any work for the award of any other degree.

Date:

SRIMANTH M. **245320748045**

ACKNOWLEDGEMENT

I am happy to express our deep sense of gratitude to the principal of the college **Dr. R. Shyam Sunder, Professor**, Neil Gogte Institute of Technology, for having provided us with adequate facilities to pursue our project.

I would like to thank**, Dr. K. Madhuri**, **Head of the Department,** CSE(AIML), Neil Gogte Institute of Technology, for having provided the freedom to use all the facilities available in the department, especially the laboratories and the library.

I would also like to thank my internal guide **Mrs. M. Deepika, Assistant Professor** for her technical guidance & constant encouragement.

I sincerely thank my seniors and all the teaching and non-teaching staff of the Department of Computer Science & Engineering for their timely suggestions, healthy criticism and motivation during this work.

Finally, I express my immense gratitude with pleasure to the other individuals who have either directly or indirectly contributed to our need at the right time for the development and success of this work.

ABSTRACT

Pattern recognition (PR) is realized as a human recognition process which can be completed by computer technology. I should first enter useful information of identifying the object into the computer. For this reason, I must abstract the recognition object and establish its mathematical model to describe it and replace the recognition object for what the machine can process. The description of this object is the pattern. Simply speaking, the pattern recognition is to identify the category to which the object belongs, such as the face in face recognition. My application is based on PR which is to identify the bird's species.

TABLE OF CONTENTS

List of Figures

List of Tables

CHAPTER – 1
INTRODUCTION

1.1 PROBLEM STATEMENT

Birds as a species are extremely diverse. There a currently 10000 species of birds on our planet Earth. The very problem of classification is an extremely complex problem. There happen to be many use cases in the market for this problem statement as deforestation is increasing many species of birds are going extinct, if I find a bird which is still alive and not extinct, using this algorithm I can spread awareness regarding the species condition.

1.2 MOTIVATION

This problem is not only challenging but also its solution is applicable to other fine-grained classification problems, helping scientists save time and resources when conducting studies about the health and abundance of certain species populations. Ultimately, I found birds to be the most interesting class to experiment with due to their immense diversity, loving nature and abundance in photographs, but I also hope to expand our understanding of the fine-grained problem and provide a useful tool for scientists across disciplines.

1.3 SCOPE

This application deals with the problem of developing a bird species identification system with the use of CNN (Convolutional Neural Networks) to classify different bird species that can be made on a general usage such as (Bald Eagle, Indian Pitta, etc..). The proposed system consists of mainly three phases: the first phase (i.e., pre-processing), the next phase (i.e., model training) and the final phase (i.e., classification). The first phase includes “ImageDataGenerator” module from the “Keras” package which is used to resize the image and segregates the images

into training

and testing datasets. In the next phase, I train the model on “training” and “validation” datasets using 10 “epoch” cycles and 2208 “steps_per_epoch” as Ill as 71 “validation_steps”. The next phase, which constitutes the main part of this project, is devoted to the classification problem where the model is used to classify different bird species.

1.4 OUTLINE

The VGG16 CNN model classifies the dataset into three parts, i.e., training dataset, testing dataset and lastly, validation dataset. The model learns from the training dataset and then using the validation dataset, inputs are validated. I used “Streamlit” to create web application that is flexible to use 200MB sized images as its input.

CHAPTER – 2

LITERATURE SURVEY

EXISTING SYSTEM:

Machine learning (ML) represents a set of techniques that allow systems to discover the required representations to features detection or classification from the raw data. The performance of works in the classification system depends on the quality of the features. As such of this study can be categorize under the field of ML, this is to make a search in this area for the studies that belong to birds' identification. Also, in the field of birds' identification system, the researchers in proposed a new feature to distinguish the types of birds. In their study, they used the ratio of the distance from the eye to the beak root, and the beak width. This feature was integrated in the decision tree, and then in SVM. This proposal was applied to the database that called (CUB-200-2011 dataset) that mentioned in the results achieved for correct classification rate is 84%.

PROPOSED SYSTEM:

The Convolutional Neural Network (CNN) is a deep learning algorithm which includes an input image and assigns the weights and the distinctions to the various aspects of the images and can then distinguish one image from another. The pre-processing required in CNN compared with other classification algorithms is much lower. In primitive methods, filters Ire usually hand-engineered; on the other hand, CNN could learn these filters on its own when subjected to enough number of trainings. CNN's architecture is quite like that of the pattern of neuron connectivity in the human brain, in which individual neurons respond only to stimuli in the receptive field. These receptive areas collectively overlap the entire visual area. The initial

parameters to be known are the elements that are a significant part in the operation of Convolutional Neural Ne

- Input Image
- CNN
- Output Label (Image Class)

CHAPTER - 3

SOFTWARE REQUIREMENTS SPECIFICATION

3.1 Overall Description:

This SRS is an overview of the whole project scenario. This document is to present a detailed description of the course management system. It will explain the purpose and features of the system, the interfaces of the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both stakeholders and developers of the system.

3.2. Operating Environment:

Software Requirements:

Operating System	:	Windows 7 (Min)
Front End	:	Streamlit
Back End	:	Python
Database	:	Microsoft Excel

Hardware Requirements:

Processor	:	Intel Pentium® Dual Core Processor (Min)
Speed	:	2.9 GHz (Min)
RAM	:	2 GB (Min)
Hard Disk	:	2 GB (Min)

3.3 Functional Requirements:

User Functionality:

- The user will be able to upload images regarding the species he/she wants to find.
- The user will be able to insert images up to 200MB in size.
- The user can see information regarding the species such as its genus, its distinguishing features, for example, the beak of an eagle.

Admin Functionality:

- The admin manages the website.
- The admin can increase database size.
- The admin can make changes to the website such as modifying the UI and making it more interactive than earlier.
- The admin can implement a better algorithm if at all a better algorithm is created in future.

3.4 Non-Functional Requirements:

3.4.1 Performance Requirements:

Performance requirements refer to static numerical requirements placed on the interaction betIen the users and the software.

Response Time:

Average response time shall be less than 5 sec.

Recovery Time:

In case of system failure, the redundant system shall resume operations within 30 secs. Average repair time shall be less than 45 minutes.

Start-Up/Shutdown Time:

The system shall be operational within 1 minute of starting up.

Capacity:

The system accommodates 1000 Concurrent Users.

Utilization of Resources:

The system shall store in the database no more than 450 different species with room for improvement.

3.4.2 Safety Requirements:

-NA-

3.4.3 Security Requirements:

The model will be running on a secure website i.e., an HTTPS website and on a secure browser such as Google Chrome, Brave, etc.

3.4.4 Software Quality Attributes:

Reliability:

The system shall be reliable i.e., in case the webpage crashes, progress will be saved.

Availability:

The website will be available to all its users round the clock i.e., they can access the website at any time.

Security:

The model will be running on a secure website i.e., an HTTPS website and on a secure browser such as Google Chrome, Brave, etc.

Maintainability:

The model shall be designed in such a way that it will be very easy to maintain it in future. Our model is a neural network model and a web-based system and will depend much on the web server and on the neural networks. However, the web application will be designed using Streamlit which is based on neural network approach and proper database modeling along with extensive documentation which

will make it easy to develop, troubleshoot and maintain in future.

Usability:

The interfaces of the system will be user friendly enough that every user will be able to use it easily.

Scalability:

The system will be designed in such a way that it will be extendable. If more species or algorithms are going to be added in the system, then it would easily be done.

The same system can also be developed to become a mobile application rather than just a website.

CHAPTER-4

SYSTEM DESIGN

Use case Diagram:

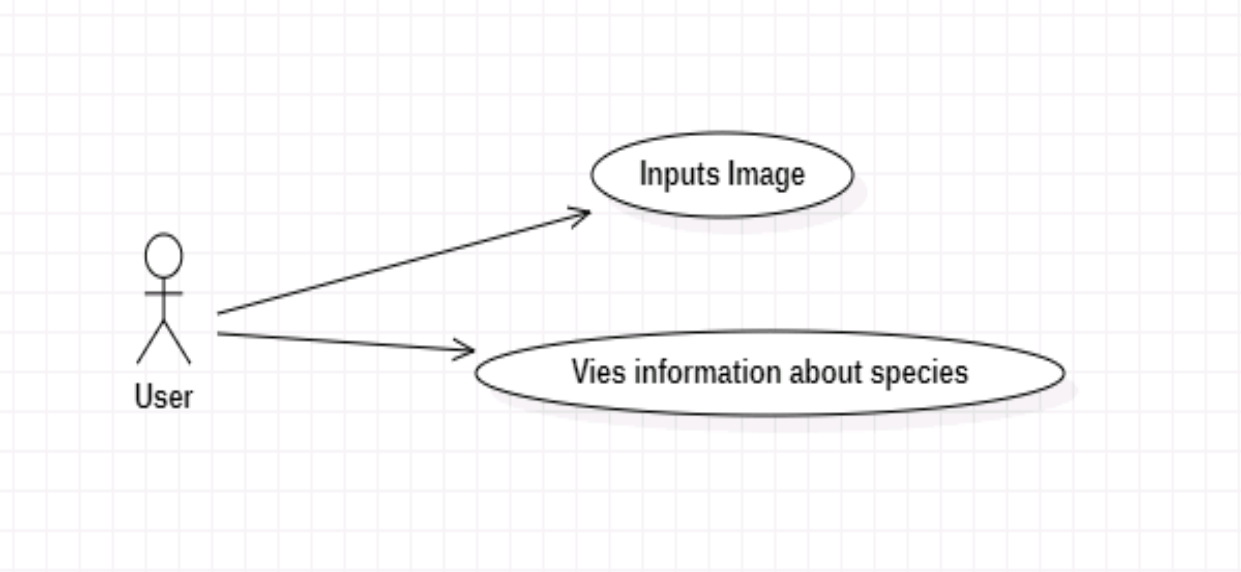

Fig 4.1: Use case diagram for User

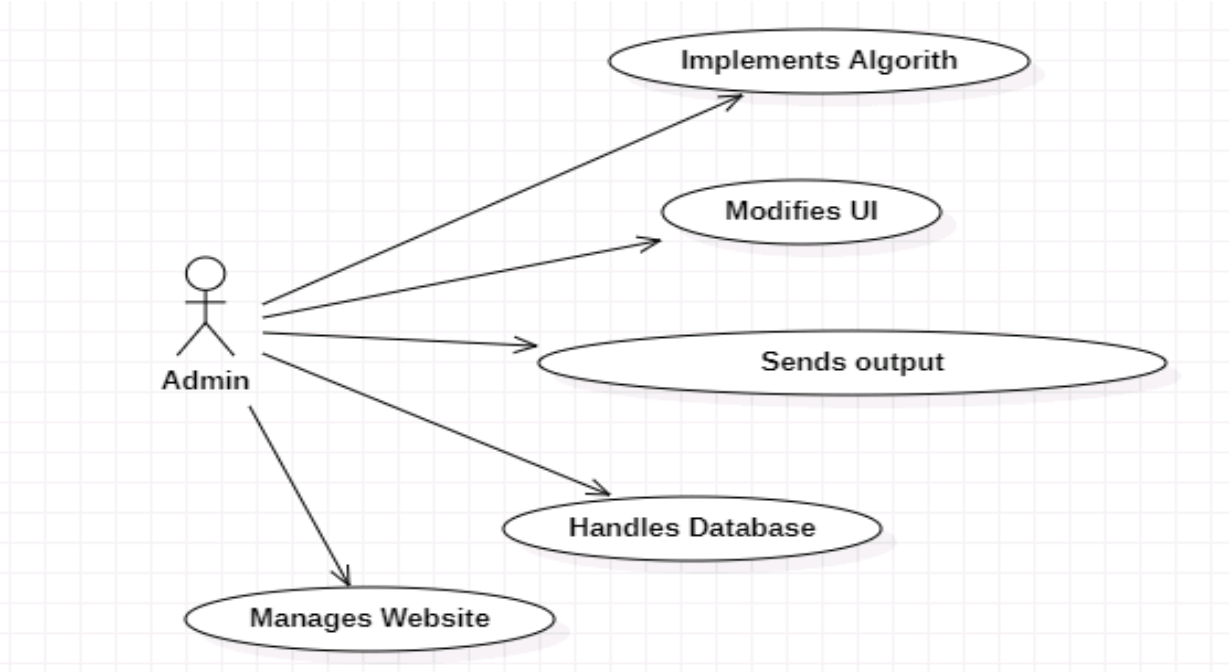

Fig 4.2: Use case diagram for Admin

Class Diagram:

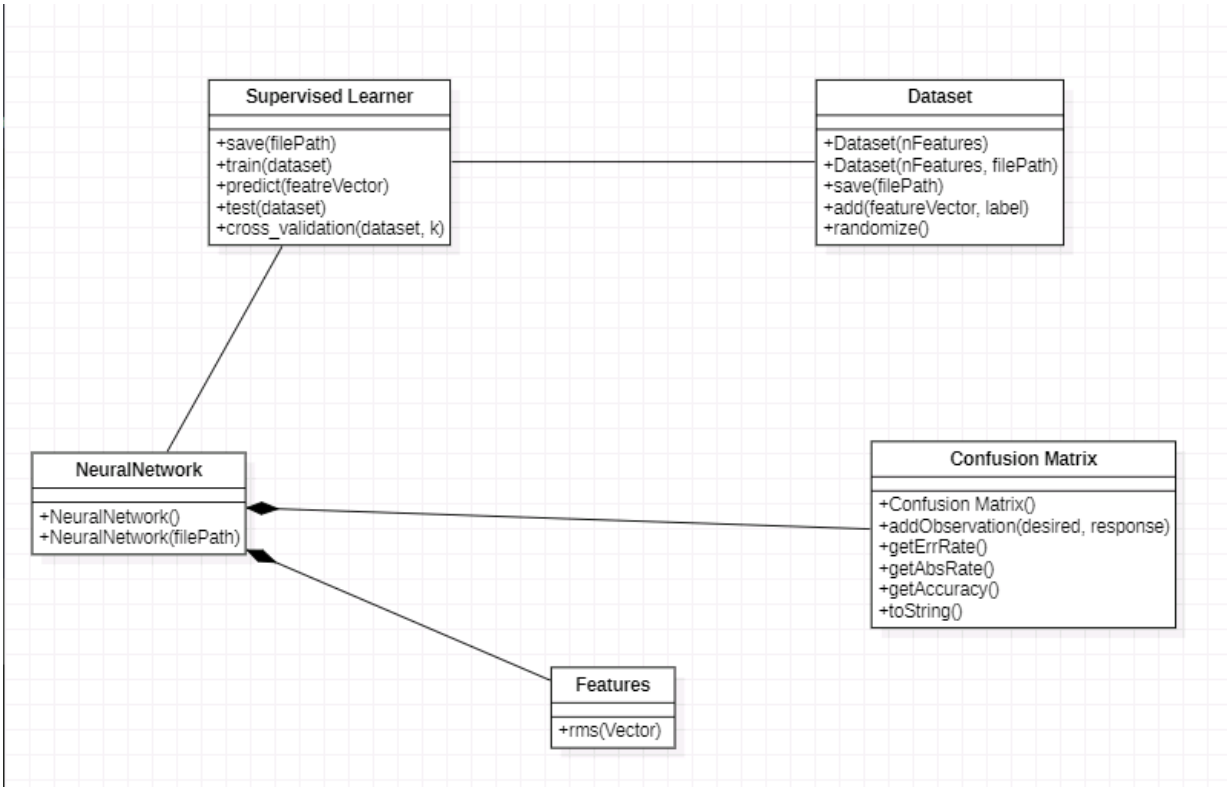

Fig.4.3 : Class diagram for Species Identification.

Sequence Diagram:

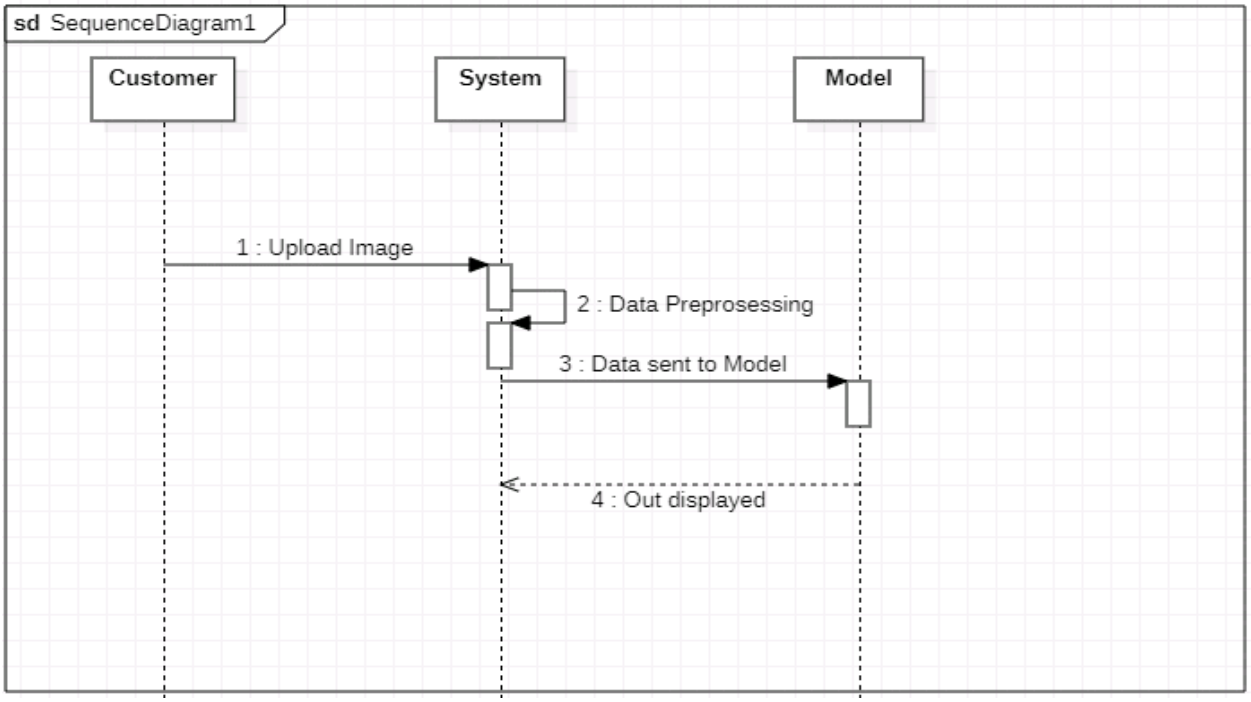

Fig4.3 : Sequence Diagam for Species Identification.

CHAPTER – 5

IMPLEMENTATION

5.1 SAMPLE CODE

```
# In[1]:

import math

from keras.layers import Input, Lambda, Dense, Flatten

from keras.models import Model

from keras.applications.vgg16 import VGG16

from keras.applications.vgg16 import preprocess_input

from keras.preprocessing import image

from keras.preprocessing.image import ImageDataGenerator

from keras.models import Sequential

import numpy as np

from glob import glob

import matplotlib.pyplot as plt

import tensorflow as tf
```

```
import keras as k




print(tf.__version__)

print(k.__version__)

# In[2]:

IMAGE_SIZE = [224, 224]

# In[3]:

train_directory='C:/Users/Srimanth/OneDrive/Desktop/Projects/Mini
Project/StreamLit/Bird_Species_Classification_Streamlit-master/train'

test_directory='C:/Users/Srimanth/OneDrive/Desktop/Projects/Mini
Project/StreamLit/Bird_Species_Classification_Streamlit-master/test'

val_directory='C:/Users/Srimanth/OneDrive/Desktop/Projects/Mini
Project/StreamLit/Bird_Species_Classification_Streamlit-master/valid'

# In[4]:

# add preprocessing layer to the front of VGG

vgg = VGG16(input_shape=IMAGE_SIZE + [3], Iights='imagenet',
include_top=False)

# don't train existing Iights

for layer in vgg.layers:

  layer.trainable = False

# In[5]:
```

```
folders = glob('C:/Users/Srimanth/OneDrive/Desktop/Projects/Mini Project/StreamLit/Bird_Species_Classification_Streamlit-master/train/*')



len(folders)

# In[6]:

# our layers - you can add more if you want

x = Flatten()(vgg.output)

# x = Dense(1000, activation='relu')(x)

prediction = Dense(len(folders), activation='softmax')(x)

# In[7]:

# create a model object

model = Model(inputs=vgg.input, outputs=prediction)

# view the structure of the model

model.summary()

# In[8]:

model.compile(

  loss='categorical_crossentropy',

  optimizer='adam',

metrics=['accuracy']

)
```

In[9]:

```
from keras.preprocessing.image import ImageDataGenerator
train_datagen = ImageDataGenerator(rescale = 1./255,
                                   shear_range = 0.2,
                                   zoom_range = 0.2,
                                   horizontal_flip = True)
test_datagen = ImageDataGenerator(rescale = 1./255)
training_set = train_datagen.flow_from_directory(train_directory,
                                                 target_size = (224, 224),
                                                 batch_size = 32,
                                                 class_mode = 'categorical')
test_set = test_datagen.flow_from_directory(test_directory,
                                            target_size = (224, 224),
                                            batch_size = 32,
                                            class_mode = 'categorical')
```

In[10]:

```
print(len(training_set))
print(len(test_set))
```

```
print(len(training_set)/len(test_set))
```

```
# In[11]:

r = model.fit(
  training_set,
  validation_data=test_set,
  epochs=10,
  steps_per_epoch=len(training_set),
  validation_steps=len(test_set)
)
# # loss
# plt.plot(r.history['loss'], label='train loss')
# plt.plot(r.history['val_loss'], label='val loss')
# plt.legend()
# plt.show()
# In[12]:
# accuracies
plt.plot(r.history['accuracy'], label='train acc')
plt.plot(r.history['val_accuracy'], label='val acc')
```

```
plt.legend()

plt.show()



# In[13]:

model.save('BC.h5')

# In[14]:

from tensorflow.keras.utils import load_img

from tensorflow.keras.utils import img_to_array

from keras import models

model1 = models.load_model('./BC.h5',compile=False)

# In[15]:

lab = training_set.class_indices

lab={k:v for v,k in lab.items()}

# In[16]:

def output(location):

    img=load_img(location,target_size=(224,224,3))

    img=img_to_array(img)

    img=img/255

    img=np.expand_dims(img,[0])

    ansIr=model1.predict(img)
```

```
    y_class = ansIr.argmax(axis=-1)
    print(y_class)



  y = " ".join(str(x) for x in y_class)
    y = int(y)
    res = lab[y]
    return res
# In[1]:
img='C:/Users/Srimanth/OneDrive/Desktop/test1.jpg'
pic=load_img(img,target_size=(224,224,3))
plt.imshow(pic)
output(img)
```

temp.py:

```
import streamlit as st
from PIL import Image
from tensorflow import keras
from tensorflow.keras.utils import load_img
from tensorflow.keras.utils import img_to_array
from keras import models
import numpy as np
```

model = models.load_model('C:/Users/Srimanth/OneDrive/Desktop/Projects/Mini Project/StreamLit/Bird_Species_Classification_Streamlit-master/model/BC.h5',compile=False)

lab ={0 :'ABBOTTS BABBLER',1 : 'ABBOTTS BOOBY',2 : 'ABYSSINIAN GROUND HORNBILL',3 : 'AFRICAN CROWNED CRANE',4 :'AFRICAN EMERALD CUCKOO',5 : 'AFRICAN FIREFINCH',6 : 'AFRICAN OYSTER CATCHER',7 : 'AFRICAN PIED HORNBILL',8 : 'ALBATROSS',9 : 'ALBERTS TOWHEE',10 : 'ALEXANDRINE PARAKEET',11 : 'ALPINE CHOUGH',12 : 'ALTAMIRA YELLOWTHROAT',13 : 'AMERICAN AVOCET',14 :'AMERICAN BITTERN',15 :'AMERICAN COOT',16 : 'AMERICAN FLAMINGO',17 : 'AMERICAN GOLDFINCH',18 : 'AMERICAN KESTREL',19 : 'AMERICAN PIPIT',20 : 'AMERICAN REDSTART',21 : 'AMERICAN WIGEON',22 : 'AMETHYST WOODSTAR',23 : 'ANDEAN GOOSE',24 : 'ANDEAN LAPWING',25 : 'ANDEAN SISKIN',26 : 'ANHINGA',27 : 'ANIANIAU',28 : 'ANNAS HUMMINGBIRD',29 : 'ANTBIRD',30 : 'ANTILLEAN EUPHONIA',31 : 'APAPANE',32 : 'APOSTLEBIRD',33 :'ARARIPE MANAKIN',34 : 'ASHY STORM PETREL',35 : 'ASHY THRUSHBIRD',36 : 'ASIAN CRESTED IBIS',37 : 'ASIAN DOLLARD BIRD',38 : 'AUCKLAND SHAQ',39 : 'AUSTRAL CANASTERO',40 : 'AUSTRALASIAN FIGBIRD',41 : 'AVADAVAT',42 : 'AZARAS SPINETAIL',43 : 'AZURE BREASTED PITTA',44 : 'AZURE JAY',45 : 'AZURE TANAGER',46 : 'AZURE TIT',47 : 'BAIKAL TEAL',48 : 'BALD EAGLE',49 : 'BALD IBIS',50 : 'BALI STARLING',51 : 'BALTIMORE ORIOLE',52 : 'BANANAQUIT',53 : 'BAND TAILED GUAN',54 :'BANDED

BROADBILL',55 : 'BANDED PITA',56 : 'BANDED STILTslide',57 : 'BARN OWL',58 : 'BARN SWALLOW',59 : 'BARRED PUFFBIRD',60 : 'BARROWS GOLDENEYE',61 : 'BAR-TAILED GODWIT',62 : 'BAY-BREASTED WARBLER',63 : 'BEARDED BARBET',64 : 'BEARDED

BELLBIRD',65 : 'BEARDED REEDLING',66 :

'BELTED KINGFISHER',67 : 'BIRD OF PARADISE',68 : 'BLACK & YELLOW BROADBILL',69 : 'BLACK BAZA',70 : 'BLACK COCKATO',71 : 'BLACK FRANCOLIN',72 : 'BLACK SKIMMER',73 : 'BLACK SWAN',74 : 'BLACK TAIL CRAKE',75 : 'BLACK THROATED BUSHTIT',76 : 'BLACK THROATED WARBLER',77 : 'BLACK VENTED SHEARWATER',78 : 'BLACK VULTURE',79 : 'BLACKBURNIAM WARBLER',80 : 'BLACK-CAPPED CHICKADEE',81 : 'BLACK-NECKED GREBE',82 : 'BLACK-THROATED SPARROW',83 : 'BLONDE CRESTED WOODPECKER',84 : 'BLOOD PHEASANT',85 : 'BLUE COAU',86 : 'BLUE DACNIS',87 : 'BLUE GROUSE',88 :'BLUE HERON',89 : 'BLUE MALKOHA',90 : 'BLUE THROATED TOUCANET',91 : 'BOBOLINK',92 : 'BORNEAN BRISTLEHEAD',93 : 'BORNEAN LEAFBIRD',94 : 'BORNEAN PHEASANT',95 : 'BRANDT CORMARANT',96 : 'BREIRS BLACKBIRD',97 : 'BROWN CREPPER',98 : 'BROWN NOODY',99 : 'BROWN THRASHER',100 :'BUFFLEHEAD',101 : 'BULIRS PHEASANT',102 : 'BURCHELLS COURSER',103 : 'BUSH TURKEY',104 : 'CAATINGA CACHOLOTE',105 : 'CACTUS WREN',106 : 'CALIFORNIA CONDOR',107 : 'CALIFORNIA GULL',108 : 'CALIFORNIA QUAIL',109 : 'CAMPO FLICKER',110 : 'CANARY',111 : 'CAPE GLOSSY STARLING',112 : 'CAPE LONGCLAW',113 : 'CAPE MAY WARBLER',114 : 'CAPE ROCK THRUSH',115 : 'CAPPED HERON',116 :'CAPUCHINBIRD',117 : 'CARMINE BEE-EATER',118 : 'CASPIAN TERN',119 :

'CASSOWARY',120 :'CEDAR WAXWING',121 : 'CERULEAN WARBLER',122 : 'CHARA DE COLLAR',123 :'CHATTERING LORY',124 :

'CHESTNET BELLIED EUPHONIA',125 : 'CHINESE BAMBOO PARTRIDGE',126 : 'CHINESE POND HERON',127 : 'CHIPPING

SPARROW',128 : 'CHUCAO TAPACULO',129 : 'CHUKAR PARTRIDGE',130 : 'CINNAMON ATTILA',131 :'CINNAMON FLYCATCHER',132 :'CINNAMON TEAL',133 : 'CLARKS NUTCRACKER',134 : 'COCK OF THE ROCK',135 : 'COCKATOO',136 : 'COLLARED ARACARI',137 : 'COMMON FIRECREST',138 : 'COMMON GRACKLE',139 : 'COMMON HOUSE MARTIN',140 : 'COMMON IORA',141 : 'COMMON LOON',142 : 'COMMON POORWILL',143 : 'COMMON STARLING',144 : 'COPPERY TAILED COUCAL',145 : 'CRAB PLOVER',146 : 'CRANE HAWK',147 : 'CREAM COLORED WOODPECKER',148 : 'CRESTED AUKLET',149 : 'CRESTED CARACARA',150 : 'CRESTED COUA',151 : 'CRESTED FIREBACK',152 : 'CRESTED KINGFISHER',153 : 'CRESTED NUTHATCH',154 : 'CRESTED OROPENDOLA',155 : 'CRESTED SHRIKETIT',156 : 'CRIMSON CHAT',157 : 'CRIMSON SUNBIRD',158 :'CROW',159 : 'CROWNED PIGEON',160 : 'CUBAN TODY',161 : 'CUBAN TROGON',162 : 'CURL CRESTED ARACURI',163 :'DALMATIAN PELICAN',164 : 'DARJEELING WOODPECKER',165 : 'DARK EYED JUNCO',166 :'D-ARNAUDS BARBET',167 : 'DARWINS FLYCATCHER',168 : 'DAURIAN REDSTART',169 : 'DEMOISELLE CRANE',170 : 'DOUBLE BARRED FINCH',171 : 'DOUBLE BRESTED CORMARANT',172 : 'DOUBLE EYED FIG PARROT',173 : 'DOWNY WOODPECKER',174 : 'DUSKY LORY',175 : 'DUSKY ROBIN',176 : 'EARED PITA',177 : 'EASTERN BLUEBIRD',178 : 'EASTERN BLUEBONNET',179 : 'EASTERN GOLDEN IAVER',180 : 'EASTERN MEADOWLARK',181 : 'EASTERN

ROSELLA',182 : 'EASTERN TOIE',183 : 'EASTERN WIP POOR WILL',184 : 'ECUADORIAN HILLSTAR',185 : 'EGYPTIAN GOOSE',186 :

'ELEGANT TROGON',187 : 'ELLIOTS PHEASANT',188 :

'EMERALD TANAGER',189 : 'EMPEROR PENGUIN',190 :'EMU',191 : 'ENGGANO MYNA',192 : 'EURASIAN BULLFINCH',193 : 'EURASIAN GOLDEN ORIOLE',194 : 'EURASIAN MAGPIE',195 : 'EUROPEAN GOLDFINCH',196 : 'EUROPEAN TURTLE DOVE',197 : 'EVENING GROSBEAK',198 : 'FAIRY BLUEBIRD',199 : 'FAIRY PENGUIN',200 : 'FAIRY TERN',201 : 'FAN TAILED WIDOW',202 : 'FASCIATED WREN',203 : 'FIERY MINIVET',204 : 'FIORDLAND PENGUIN',205 : 'FIRE TAILLED MYZORNIS',206 : 'FLAME BOIRBIRD',207 : 'FLAME TANAGER',208 : 'FRIGATE',209 : 'GAMBELS QUAIL',210 : 'GANG GANG COCKATOO',211 : 'GILA WOODPECKER',212 : 'GILDED FLICKER',213 : 'GLOSSY IBIS',214 : 'GO AWAY BIRD',215 : 'GOLD WING WARBLER',216 : 'GOLDEN BOIR BIRD',217 : 'GOLDEN CHEEKED WARBLER',218 : 'GOLDEN CHLOROPHONIA',219 : 'GOLDEN EAGLE',220 : 'GOLDEN PARAKEET',221 : 'GOLDEN PHEASANT',222 : 'GOLDEN PIPIT',223 : 'GOULDIAN FINCH',224 : 'GRANDALA',225 : 'GRAY CATBIRD',226 : 'GRAY KINGBIRD',227 : 'GRAY PARTRIDGE',228 : 'GREAT GRAY OWL',229 : 'GREAT JACAMAR',230 : 'GREAT KISKADEE',231 : 'GREAT POTOO',232 : 'GREAT TINAMOU',233 : 'GREAT XENOPS',234 : 'GREATER PEIE',235 : 'GREATOR SAGE GROUSE',236 : 'GREEN BROADBILL',237 : 'GREEN JAY',238 : 'GREEN MAGPIE',239 : 'GREY CUCKOOSHRIKE',240 : 'GREY PLOVER',241 : 'GROVED BILLED ANI',242 : 'GUINEA TURACO',243 : 'GUINEAFOWL',244 : 'GURNEYS PITTA',245 : 'GYRFALCON',246 : 'HAMERKOP',247 : 'HARLEQUIN DUCK',248 : 'HARLEQUIN QUAIL',249 : 'HARPY EAGLE',250 : 'HAWAIIAN

GOOSE',251 :'HAWFINCH',252 : 'HELMET VANGA',253 : 'HEPATIC TANAGER',254 : 'HIMALAYAN BLUETAIL',255 : 'HIMALAYAN MONAL',256 : 'HOATZIN',257 : 'HOODED MERGANSER',258 : 'HOOPOES',259 : 'HORNED GUAN',260 : 'HORNED LARK',261 : 'HORNED SUNGEM',262 : 'HOUSE FINCH',263 : 'HOUSE SPARROW',264 : 'HYACINTH MACAW',265 :'IBERIAN MAGPIE',266 : 'IBISBILL',267 : 'IMPERIAL SHAQ',268 : 'INCA TERN',269 : 'INDIAN BUSTARD',270 : 'INDIAN PITTA',271 : 'INDIAN ROLLER',272 : 'INDIAN VULTURE',273 : 'INDIGO BUNTING',274 : 'INDIGO FLYCATCHER',275 : 'INLAND DOTTEREL',276 :'IVORY BILLED ARACARI',277 : 'IVORY GULL',278 : 'IWI',279 : 'JABIRU',280 : 'JACK SNIPE',281 : 'JANDAYA PARAKEET',282 : 'JAPANESE ROBIN',283 : 'JAVA SPARROW',284 : 'JOCOTOCO ANTPITTA',285 : 'KAGU',286 : 'KAKAPO',287 : 'KILLDEAR',288 : 'KING EIDER',289 : 'KING VULTURE',290 : 'KIWI',291 : 'KOOKABURRA',292 : 'LARK BUNTING',293 : 'LAZULI BUNTING',294 : 'LESSER ADJUTANT',295 : 'LILAC ROLLER',296 : 'LITTLE AUK',297 : 'LOGGERHEAD SHRIKE',298 : 'LONG-EARED OWL',299 : 'MAGPIE GOOSE',300 : 'MALABAR HORNBILL',301 : 'MALACHITE KINGFISHER',302 : 'MALAGASY WHITE EYE',303 : 'MALEO',304 : 'MALLARD DUCK',305 : 'MANDRIN DUCK',306 : 'MANGROVE CUCKOO',307 : 'MARABOU STORK',308 : 'MASKED BOOBY',309 : 'MASKED LAPWING',310 : 'MCKAYS BUNTING',311 : 'MIKADO PHEASANT',312 : 'MOURNING DOVE',313 : 'MYNA',314 : 'NICOBAR PIGEON',315 : 'NOISY FRIARBIRD',316 : 'NORTHERN BEARDLESS TYRANNULET',317 : 'NORTHERN CARDINAL',318 : 'NORTHERN FLICKER',319 : 'NORTHERN FULMAR',320 : 'NORTHERN GANNET',321 : 'NORTHERN GOSHAWK',322 : 'NORTHERN JACANA',323 :

'NORTHERN MOCKINGBIRD',324 : 'NORTHERN PARULA',325 :

'NORTHERN RED BISHOP',326 : 'NORTHERN SHOVELER',327 : 'OCELLATED TURKEY',328 : 'OKINAWA RAIL',329 : 'ORANGE BRESTED BUNTING',330 : 'ORIENTAL BAY OWL',331 : 'OSPREY',332 : 'OSTRICH',333 : 'OVENBIRD',334 : 'OYSTER CATCHER',335 : 'PAINTED BUNTING',336 : 'PALILA',337 : 'PARADISE TANAGER',338 : 'PARAKETT AKULET',339 : 'PARUS MAJOR',340 : 'PATAGONIAN SIERRA FINCH',341 : 'PEACOCK',342 : 'PEREGRINE FALCON',343 : 'PHILIPPINE EAGLE',344 : 'PINK ROBIN',345 : 'POMARINE JAEGER',346 : 'PUFFIN',347 : 'PURPLE FINCH',348 : 'PURPLE GALLINULE',349 : 'PURPLE MARTIN',350 : 'PURPLE SWAMPHEN',351 : 'PYGMY KINGFISHER',352 : 'QUETZAL',353 : 'RAINBOW LORIKEET',354 : 'RAZORBILL',355 : 'RED BEARDED BEE EATER',356 : 'RED BELLIED PITTA',357 : 'RED BROID FINCH',358 : 'RED FACED CORMORANT',359 : 'RED FACED WARBLER',360 : 'RED FODY',361 : 'RED HEADED DUCK',362 : 'RED HEADED WOODPECKER',363 : 'RED HONEY CREEPER',364 : 'RED NAPED TROGON',365 : 'RED TAILED HAWK',366 : 'RED TAILED THRUSH',367 : 'RED WINGED BLACKBIRD',368 : 'RED WISKERED BULBUL',369 : 'REGENT BOIRBIRD',370 : 'RING-NECKED PHEASANT',371 : 'ROADRUNNER',372 : 'ROBIN',373 : 'ROCK DOVE',374 : 'ROSY FACED LOVEBIRD',375 : 'ROUGH LEG BUZZARD',376 : 'ROYAL FLYCATCHER',377 : 'RUBY THROATED HUMMINGBIRD',378 : 'RUDY KINGFISHER',379 : 'RUFOUS KINGFISHER',380 : 'RUFUOS MOTMOT',381 : 'SAMATRAN THRUSH',382 : 'SAND MARTIN',383 : 'SANDHILL CRANE',384 : 'SATYR TRAGOPAN',385 : 'SCARLET CROWNED FRUIT DOVE',386 : 'SCARLET IBIS',387 : 'SCARLET MACAW',388 : 'SCARLET TANAGER',389 : 'SHOEBILL',390 : 'SHORT BILLED DOWITCHER',391 : 'SKUA',392 : 'SMITHS LONGSPUR',393 :

'SNOWY EGRET',394 :

'SNOWY OWL',395 :'SNOWY PLOVER',396 : 'SORA',397 : 'SPANGLED COTINGA',398 : 'SPLENDID WREN',399 : 'SPOON BILED SANDPIPER',400 : 'SPOONBILL',401 : 'SPOTTED CATBIRD',402 : 'SRI LANKA BLUE MAGPIE',403 : 'STEAMER DUCK',404 : 'STORK BILLED KINGFISHER',405 : 'STRAWBERRY FINCH',406 : 'STRIPED OWL',407 : 'STRIPPED MANAKIN',408 : 'STRIPPED SWALLOW',409 : 'SUPERB STARLING',410 : 'SWINHOES PHEASANT',411 : 'TAILORBIRD',412 : 'TAIWAN MAGPIE',413 : 'TAKAHE',414 : 'TASMANIAN HEN',415 : 'TEAL DUCK',416 : 'TIT MOUSE',417 :'TOUCHAN',418 : 'TOWNSENDS WARBLER',419 : 'TREE SWALLOW',420 : 'TRICOLORED BLACKBIRD',421 :'TROPICAL KINGBIRD',422 : 'TRUMPTER

SWAN',423 : 'TURKEY VULTURE',424 : 'TURQUOISE MOTMOT',425 : 'UMBRELLA BIRD',426 : 'VARIED THRUSH',427 : 'VEERY',428 : 'VENEZUELIAN TROUPIAL',429 :'VERMILION FLYCATHER',

430 : 'VICTORIA CROWNED PIGEON',431 : 'VIOLET GREEN SWALLOW',432 : 'VIOLET TURACO',433 : 'VULTURINE GUINEAFOWL',434 : 'WALL CREAPER',435 : 'WATTLED CURASSOW',436 : 'WATTLED LAPWING',437 : 'WHIMBREL',438 : 'WHITE BROID CRAKE',439 : 'WHITE CHEEKED TURACO',440 : 'WHITE CRESTED HORNBILL',441 : 'WHITE NECKED RAVEN',442 : 'WHITE TAILED TROPIC',443 : 'WHITE THROATED BEE EATER',444 : 'WILD TURKEY',445 : 'WILSONS BIRD OF PARADISE',446 : 'WOOD DUCK',447 : 'YELLOW BELLIED FLOIRPECKER',448 : 'YELLOW CACIQUE449',449: 'YELLOW HEADED BLACKBIRD'}

```
def processed_img(img_path):
 img=load_img(img_path,target_size=(224,224,3))
    img=img_to_array(img)
    img=img/255
    img=np.expand_dims(img,[0])
    ansIr=model.predict(img)
    y_class = ansIr.argmax(axis=-1)
    print(y_class)
    y = " ".join(str(x) for x in y_class)
    y = int(y)
    res = lab[y]
 print(res)
    return res
def run():
        img1 = Image.open('C:/Users/Srimanth/OneDrive/Desktop/Projects/Mini
Project/StreamLit/Bird_Species_Classification_Streamlit-master/meta/logo1.png')
    img1 = img1.resize((350,350))
    st.image(img1,use_column_width=False)
```

```
    st.title("Birds Species Classification")



st.markdown("'<h4 style='text-align: left; color: #d73b5c;'>* Data is based on "450
Bird Species"</h4>'",

            unsafe_allow_html=True)

    img_file = st.file_uploader("Choose an Image of Bird", type=["jpg", "png"])

    if img_file is not None:

        st.image(img_file,use_column_width=False)

            save_image_path  =  'C:/Users/Srimanth/OneDrive/Desktop/Projects/Mini
Project/StreamLit/Bird_Species_Classification_Streamlit-
master/test'+img_file.name

        with open(save_image_path, "wb") as f:

            f.write(img_file.getbuffer())

 if st.button("Predict"):

            result = processed_img(save_image_path)

            st.success("Predicted Bird is: "+result)

run()
```

CHAPTER – 6

TESTING

6.1 TEST CASES

Test Case ID:	1
Test Case Name:	Required Software Testing
Purpose:	To check whether the required Software is installed on the systems
Input:	Enter python command
Expected Result:	Should Display the version number for the python
Actual Result:	Displays python version
Failure	If the python environment is not installed, then the Deployment fails

Test Case to check whether the required Software is installed on the systems

Table 6.1.1 python Installation verification

Test Case to check Program Integration Testing

Test Case ID:	2
Test Case Name:	Programs Integration Testing
Purpose:	To ensure that all the modules work together
Input:	All the modules should be accessed.
Expected Result:	All the modules should be functioning properly.
Actual Result:	All the modules should be functioning properly.
Failure	If any module fails to function properly, the implementation fails.

Table 6.1.2 python Programs Integration Testing

Test Case to Collect Dataset and Load the Dataset

Test Case ID:	3
Test Case Name:	Collect Dataset and Load the Dataset
Purpose:	Check Dataset is collected, and the data is stored
Input:	Provide Dataset as input
Expected Result:	Dataset is collected and view the Dataset and store the Dataset
Actual Result:	Load the Dataset and view the Dataset and store
Failure	If the dataset is not loaded, it will throw an error.

Table 6.1.3 Collect Dataset and Load the Dataset

Test Case to check whether the species is recognized

Test Case ID:	4
Test Case Name:	Species Recognition
Purpose:	Species Recognition using CNN
Input:	Provide dataset and input an image
Expected Result:	After Evaluation I get the species name
Actual Result:	After Evaluation I get the species name
Failure	If the data is not Evaluated, it does not display the gesture

Table 6.1.4 Species Recognition

CHAPTER - 7

SCREENSHOTS

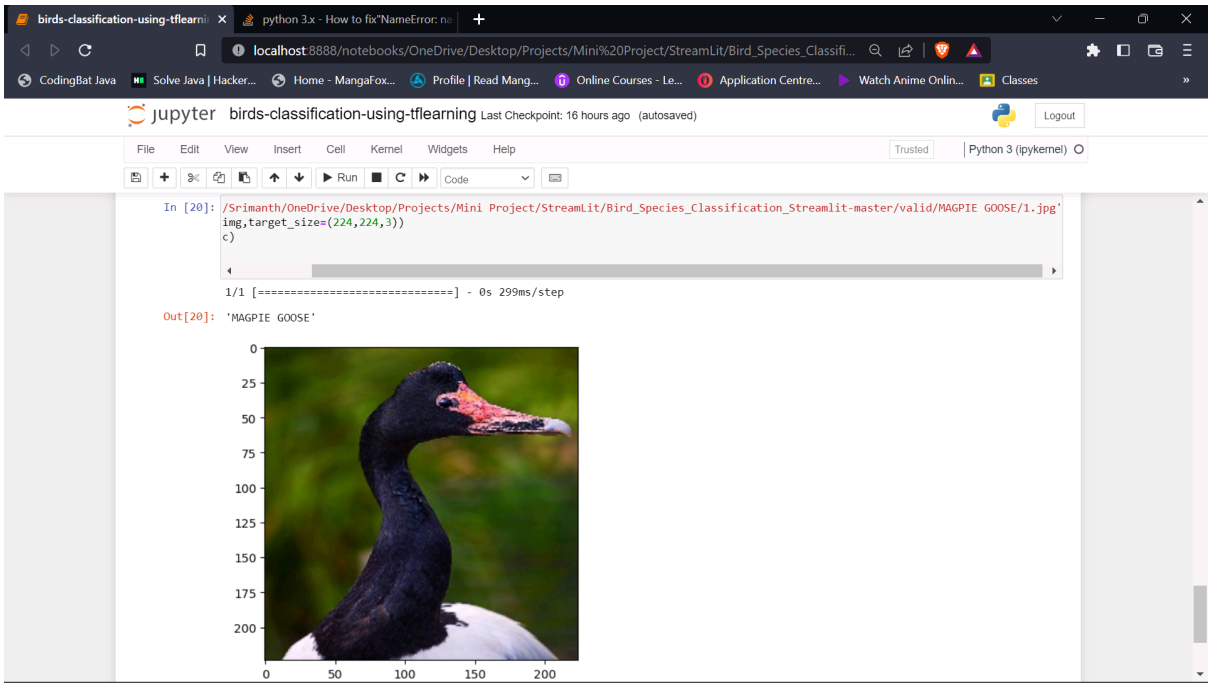

Figure 7.1:TRAINING for MAGPIE GOOSE

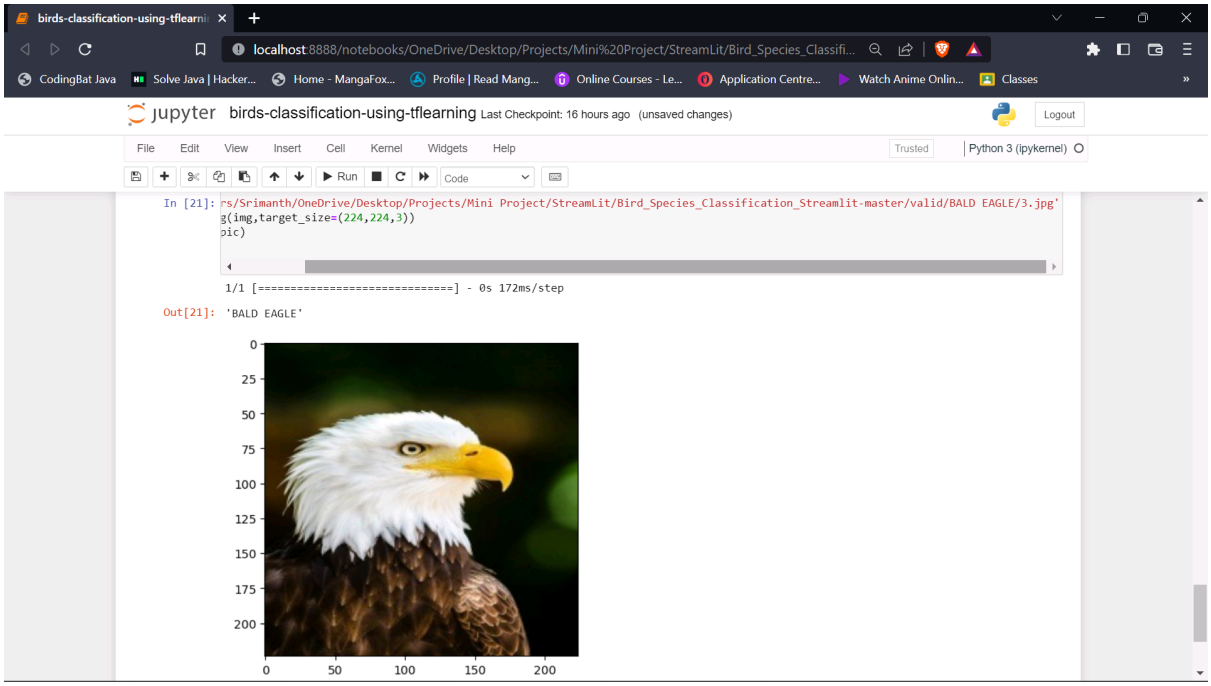

Figure 7.2: TRAINING for BALD EAGLE

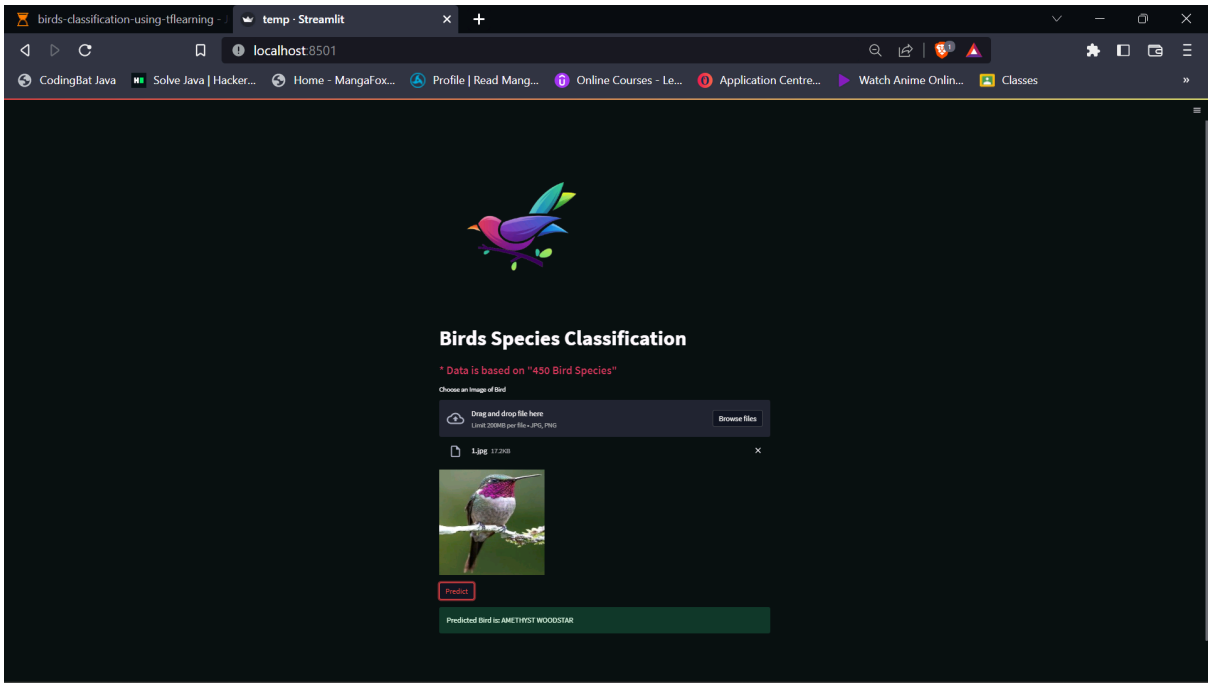

Figure 7.3: HOME PAGE for SPECIES 1

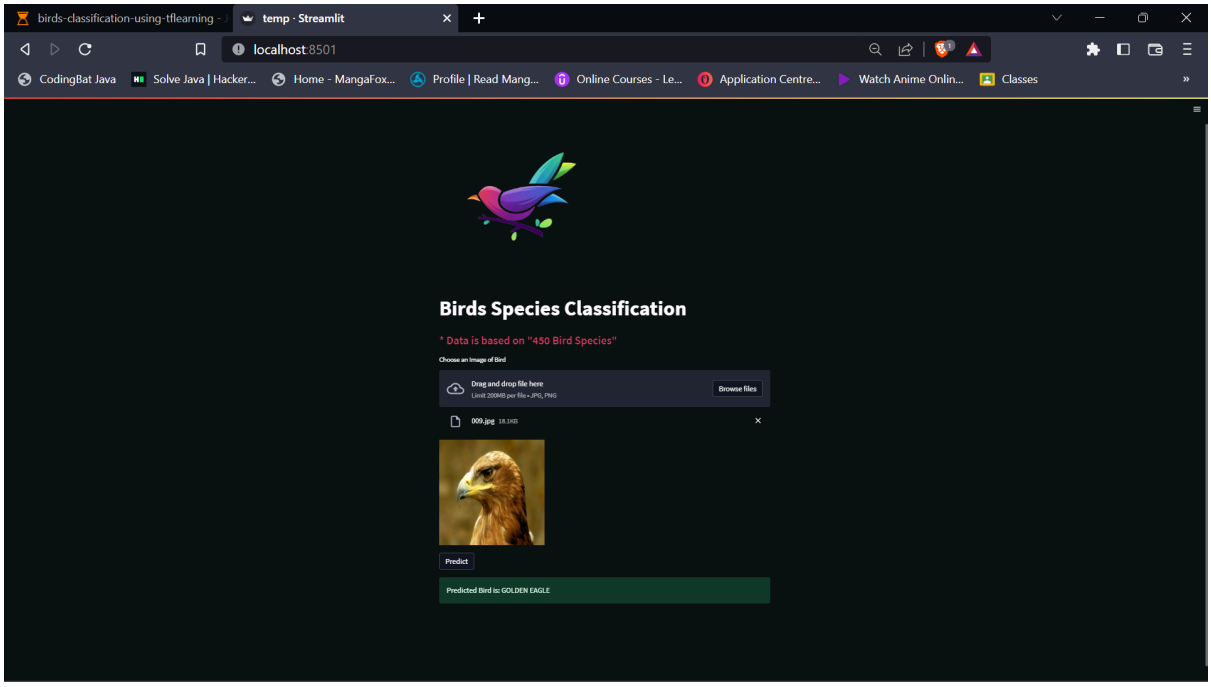

Figure 7.4: HOME PAGE for Species 2

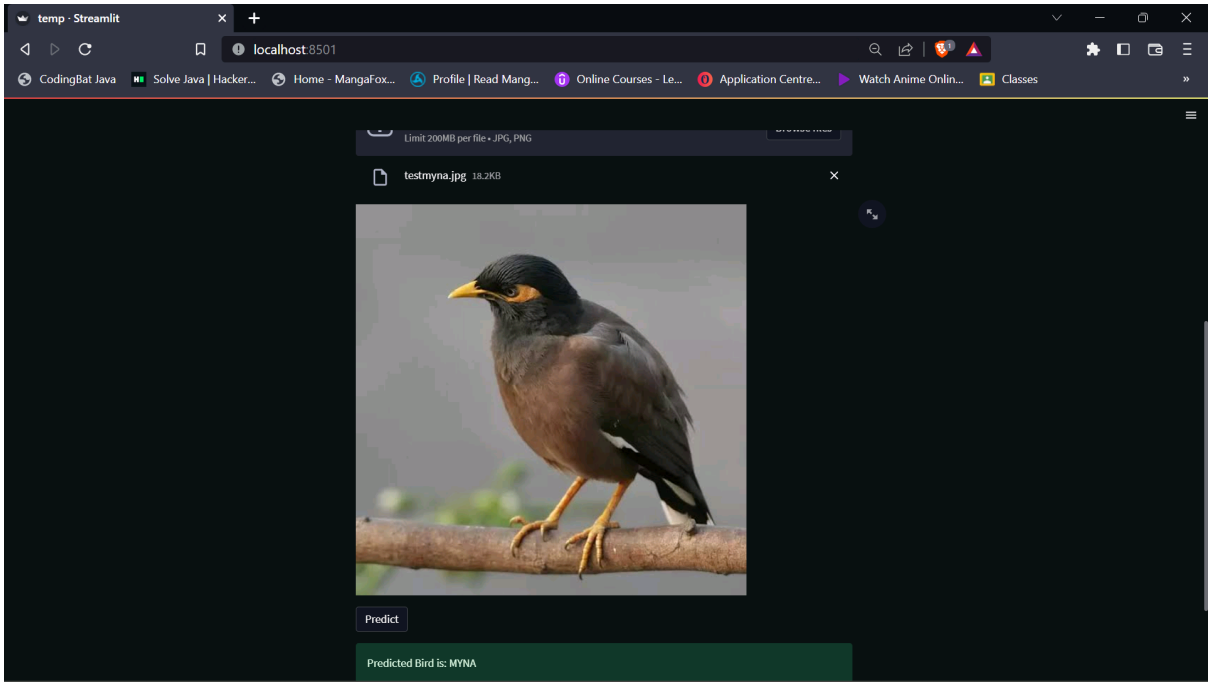

Figure 7.5: HOME PAGE for Species 3

CHAPTER - 8

CONCLUSION AND FUTURE SCOPE

The main idea behind developing the “Bird Species Identification” website is to build awareness regarding different species of birds. It also caters to the need of simplifying the bird identification process and thus making birdwatching or species identification much easier. The fundamental technology used in this project is “CNN” or “Convolutional Neural Network”.

This application is not a complete model as it has room for improvement as and when requirements appear soon. One of the major reasons why I chose this subject is, since deforestation is happening rapidly, certain species of birds may go extinct, if at all such situation appears, this application is useful in identifying such species if they Ire to be still alive. Also, the current model is based on pre-classified images, in future I intend to develop a model which can self-classify images into respective folders by using a CNN process known as “Feature Extraction” and, I intend to increase the number of “layers” in the model which makes it even more robust and future ready.

BIBLIOGRAPHY

[1] W. Man, Y. Ji, and Z. Zhang, "Image classification based on improved random forest algorithm," in 2018 IEEE 3rd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), 2018, pp. 346--350.

[2] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," ICLR, 2015.

[3] G. Amato and F. Falchi, "kNN based image classification relying on local feature similarity," in Proceedings of the Third International Conference on Similarity Search and Applications - SISAP '10, 2010, p. 101.

[4] S. Dave, "Image Classification Algorithm based on Multi-Feature Extraction and KNN Classifier," Int. J. Adv. Eng. Res. Dev., vol. 4, no. 6, 2017.

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

[6] Y. Chen, Z. Lin, and X. Zhao, "Riemannian manifold learning based k-nearest-neighbour for hyperspectral image classification," in 2013 IEEE International Geoscience and Remote Sensing Symposium - IGARSS, 2013, pp. 1975--1978.

[7] E. Blanzieri and F. Melgani, "Nearest Neighbor Classification of Remote Sensing Images with the Maximal Margin Principle," IEEE Trans. Geosci. Remote Sens., vol. 46, no. 6, pp. 1804--1811, Jun. 2008.

[8] X. Jia and J.a. Richards. 2005. Fast K-NN Classification Using the Cluster-Space Approach. IEEE Geoscience and Remote Sensing Letters2, 2 (2005), 225--228.

[9] G. Varoquaux, L. Buitinck, G. Louppe, O. Grisel, F. Pedregosa, and A. Mueller, "Scikit-learn," GetMobile Mob. Comput. Commun., vol. 19, no. 1, pp. 29--33, Jun. 2015.

[10] G. Mercier and M. Lennon, "Support vector machines for hyperspectral image classification with spectral-based kernels," in IGARSS 2003. 2003 IEEE

[11] "Birds as Environmental Indicators | Environmental Science.org." {Online}. Available: https://www.environmentalscience.org/birds-environmental-indicators. {Accessed: 04-Jun-2019}.

APPENDIX A: TOOLS AND TECHNOLOGIES

- PYTHON V3: The Python language comes with many libraries and frameworks that make coding easy. This also saves a significant amount of time.
- JUPYTER NOTEBOOK: The Jupyter Notebook is an open source web application that you can use to create and share documents that contain live code, equations, visualizations, and text.
- TENSORFLOW: TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML poIred applications.
- NUMPY: NumPy is a Python library used for working with arrays. It also has functions for working in domain of linear algebra, fourier transform, and matrices.
- WINDOWS 11: Windows 11 was used as the operating system.
- KERAS: Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library.
- VGG16: VGG-16 is a convolutional neural network that is 16 layers deep. You can load a pretrained version of the network trained on more than a million images from the ImageNet database.
- Streamlit: Streamlit is an open source app framework in Python language. It helps us create web apps for data science and machine learning in a short time. It is compatible with major Python libraries such as scikit-learn, Keras, PyTorch, SymPy(latex), NumPy, pandas, Matplotlib etc.