Kimberly Van
MSDS 462, Section 55
November 28, 2021

Final Project: Bird Species Classification App

Introduction:

The purpose of this edge-based computer vision solution is to classify bird species. The model consisted of transfer learning through using ResNet50 as the pretrained model. The goal of this computer vision solution is to assist bird watchers to quickly identify a bird species using a mobile application. The model was initially built in Python using TensorFlow in Google Colaboratory (link) and deployed onto Android using Android Studio.

Data Set

The data set used for this final project was retrieved from Kaggle's Bird Species Classification and can be accessed with the following link: https://www.kaggle.com/roniavesta/bird-species-classification/data. The data set consists of images for 300 different bird species. There were 3 folders containing images.

Folders	Total # of Images
Training Set	43622
Validation Set	1500
Test Set	1500

Each folder consisted of subfolders that were labeled by bird species type. Each subfolder consisted of ~130-150 images of the correlated bird specie in various angles. The path file and appropriate label for each image were extracted from the subfolders and stored in a dataframe using Python for the model to access.

Model:

Transfer learning model was built using ResNet50 as the pretrained model. ResNet50 is a 50-layer deep convolutional neural network architecture. This model has been trained on ImageNet's database, which included over 1 million images and across 20,000 different categories.

Deep neural networks are difficult to train due to the vanishing gradient problem. Repeated multiplication occurs as the gradient is back-propagated to earlier layers, leading to a deeper network that results in performance saturation. ResNet50 introduces a concept of skip connection, where the original input is added to the output of the convolutional block instead of stacking convolutional layers that occur in traditional neural networks. Skipping connection helps mitigate the vanishing gradient problem by allowing an alternate shortcut path for gradient to flow through. It also ensures that the higher layers will perform at least as well as the lower layers.

For this final project, two additional Dense layers were added to the ResNet50 model. The model summary is presented below:

```
Model: "sequential_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 resnet50 (Functional)       (None, 2048)              23587712

 dense_2 (Dense)             (None, 120)               245880

 dense_3 (Dense)             (None, 300)               36300

=================================================================
Total params: 23,869,892
Trainable params: 23,816,772
Non-trainable params: 53,120
_________________________________________________________________
```

The model was trained with 15 epochs that included an early stopping occurrence at 11 epochs. The training performance is presented below:

	Accuracy
Training Set	0.9529
Validation Set	0.7667

This transfer learning model outputs the percent probabilities for each species it identifies the bird. For example, an image of a barn owl can be identified as 94% barn owl, 2% Bornean pheasant, etc.

Deployment:

Android Studio was used to deploy the model into an android mobile application. The backbone of the application used was from TensorFlow's Image Classification example (link). The bird classification model file replaced the Image Classification model. The text file (labels.txt) containing the list of labels for each bird species replaced that of the original labels file. The Image Classification example was used in deploying this mobile application because of its applicable functions. Identifying bird species can be complicated. There are difficulties in capturing quality images of a bird in real life. Some bird species have similar physical characteristics. The Image Classification example in Android Studio was the most applicable for this project's application since the camera is used intake the image of the bird without having to take an actual picture of the bird.

Conclusion:

Identifying bird species can be complicated based on the difficulties of obtaining a detailed image of the bird in real life. It can also be confusing as some bird species possess similar physical characteristics. The results from the transfer learning model provides the user with a list ranking the likeliness of each identified species. One challenge with this application is the ability to capture a clear and detailed image of a bird using a smartphone. An improvement to this application would be to create a video classification app that can analyze the bird in several angles as it moves in its natural habitat.