SEASONS OF CODE PROJECT REPORT

NAME: RUPANSHI VIJAYVARGIYA

ROLL NUMBER: 23B2473

DEPARTMENT: MEMS DEPARTMENT

PROJECT NAME: DEEP LEARNING

MENTOR NAME: SHUBHAM RAJ

PROJECT TOPIC: CONVOLUTIONAL NEURAL NETWORK

CNN MODEL FOR IMAGE CLASSIFICATION OF BIRDS

• The dataset is available here: CUB\_200\_2011.zip

• It is a very large dataset, so it will get downloaded as a zipped file. Unzip this zipped file.

• Import the necessary libraries (TensorFlow)

• The image folder contains 200 subdirectories (each of which contains some image files), and these are the number of classes. The image size should be (224, 224) and the dataset should be shuffled.

• Map the data by dividing all the pixel values by 255, so that the pixel values range from 0 to 1.

• Split the image dataset with the help of train\_test\_split.txt file present in the CUB\_200\_2011 folder (the one downloaded from Data)

Introduction:

This report presents the work done for the fine-grained image classification task using a Convolutional Neural Network (CNN) model. The task involves classifying images from the CUB-200-2011 dataset into 200 bird species classes. The CNN model was designed with a parameter limit of to achieve efficient parameter utilization while maintaining good accuracy.

Dataset:

2.1 Dataset Description The CUB-200-2011 dataset contains 11788 images of 200 bird species with a diverse range of poses and backgrounds. The dataset is available for download here.

2.2 Data Split The dataset was used with its default train-test split given in the dataset. Created train and test directories according to the ‘train\_test\_split.txt’ file. Around half of the data is in the training set and the other half in test.The training set contains images for model training, while the test set is used for evaluating model performance.

Model Architecture:

3.1 Base Model The MobileNetV2 architecture was selected as the base model for this task. MobileNetV2 is a [convolutional neural network](https://www.geeksforgeeks.org/introduction-convolution-neural-network/) architecture optimized for mobile and embedded vision applications. It improves upon the original MobileNet by introducing inverted residual blocks and linear bottlenecks, resulting in higher accuracy and speed while maintaining low computational costs. MobileNetV2 is widely used for tasks like [image classification](https://www.geeksforgeeks.org/cifar-10-image-classification-in-tensorflow/), [object detection, and semantic segmentation](https://www.geeksforgeeks.org/object-detection-vs-object-recognition-vs-image-segmentation/) on mobile and edge devices.

Model Details:

● Base Model: MobileNetV2

● Input Shape: (224, 224, 3)

● Output Layer: Dense layer with 200 units and softmax activation for multi-class classification

● Total Parameters: 6,808,218

Training Details

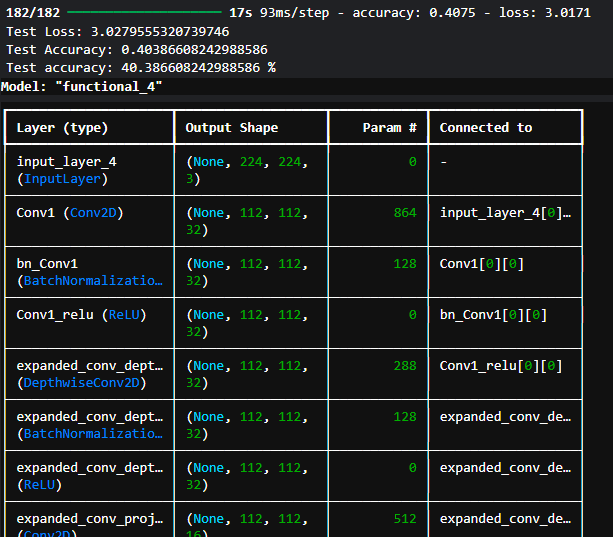
Training Parameters

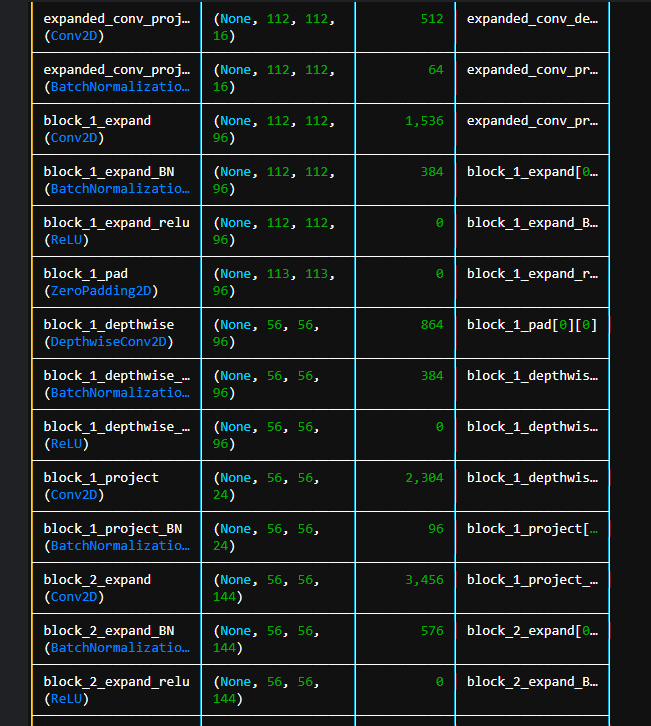
● Optimizer: Adam

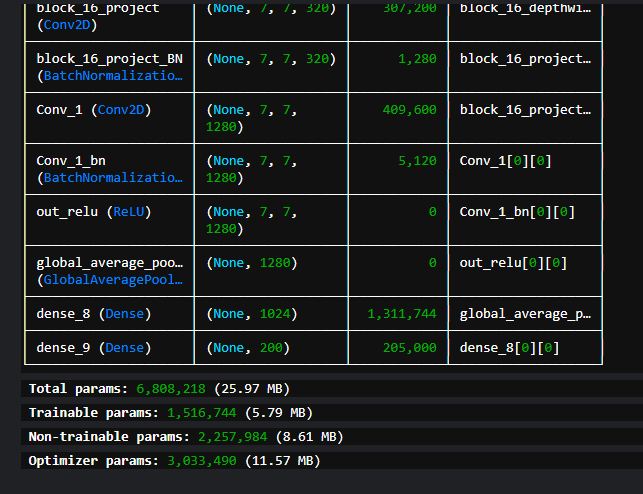
● Epochs: 10

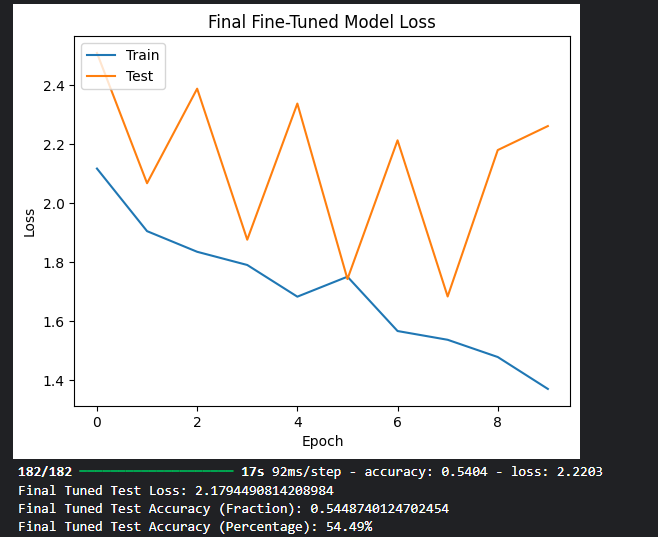
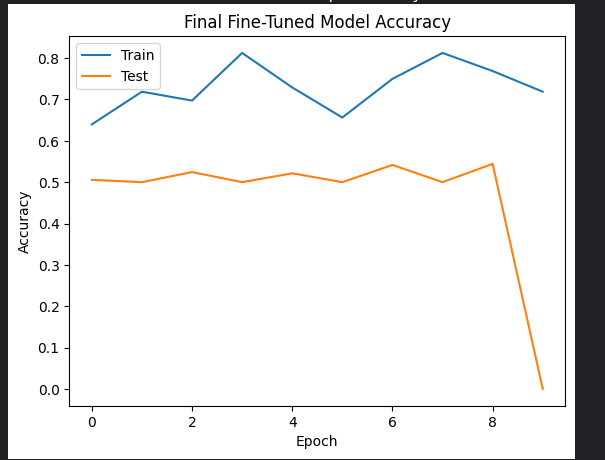
Training Process:

The model was trained using the training dataset with validation performed on the test dataset.







Training Loss and Accuracy The following plots show the training and validation loss and accuracy over epochs 

Final Results :

Model Evaluation The model was evaluated on the test dataset, achieving a final accuracy of : 54.49%