Plant Disease Classification

B. TECH SEM – VII Minor PROJECT
Dept. of Computer Science & Engineering

By

Dhyey Shah 19BCP035
Devam Jariwala 19BCP025

Under the Supervision Of: **Dr Rajeev Gupta**

SCHOOL OF TECHNOLOGY
PANDIT DEENDAYAL ENERGY UNIVERSITY
GANDHINAGAR, GUJARAT, INDIA

July – December, 2022

ABSTRACT

The majority of plant diseases have obvious symptoms, and an experienced plant pathologist will typically identify the disease by physically examining the afflicted plant leaves. Due to the lengthy manual illness diagnosis process and the fact that the accuracy of the diagnosis is inversely proportional to the pathologist's skills, this challenge is an ideal application area for computer-assisted diagnostic systems. Traditional machine learning techniques, where human feature extraction must be flawless to yield useful results, must be replaced with a model that does not require pre-processing and can successfully conduct categorization. In this study, the MobileNetV2 architecture for deep learning was proposed for classifying plant leaf diseases. The model was compared to Residual Network and its performance and irregularities were investigated.
Keywords: - Crop disease classification, Deep Learning, MobileNetV2, Residual Network

TABLE OF CONTENTS

1. INTRODUCTION: -

There are hundreds of thousands of different crops and plants in the globe, and just 120 species are available for human consumption. Generally, crops fall into two categories: food crops and cash crops. Despite this abundance, 690 million people, or 9 percent of the world's population, are hungry. It is caused by food waste and disease-related agricultural damage. There have been less advances in agricultural disease identification and prevention despite the existence of numerous programmes to reduce food waste. The timely and accurate detection of plant diseases is essential for agriculture to be sustainable and effective, as well as to avoid the waste of money and other resources. When a plant disease lacks evident symptoms, novel analysis techniques are required. In spite of this, the overwhelming.

The traditional way for diagnosing most plant diseases is for a competent plant pathologist to visually inspect the diseased plant leaves. Diverse plant diseases have a significant impact on agricultural production. The Irish potato famine of 1845–1849, which resulted in 1.2 million deaths, is a famous example. A plant pathologist must be adept in observation in order to recognise the unique symptoms of plant disease and make an accurate diagnosis. Due to the vast variety of plants, variations in the course of plant diseases induced by climate changes, and the quick spread of diseases to previously unobserved areas, even seasoned pathologists occasionally struggle to diagnose certain diseases. Farmers and academics have never stopped trying to develop a sophisticated and practical classification system for plant diseases. The polymerase chain reaction, enzyme-linked immunosorbent assay, and loop-mediated isothermal amplification are examples of highly specific and sensitive laboratory test methods for detecting diseases in plant samples. Advances in artificial intelligence technology have cleared the path for the creation of automated disease-diagnosis systems that are faster and more accurate. Today, methods utilising artificial intelligence to autonomously identify a wide variety of diseases are frequently employed. In the previous decade, numerous traditional machine learning models for the detection and classification of plant diseases were suggested. In addition, a study was conducted on the early identification and classification of sugar beet illnesses based on spectral plant indicators and Support Vector Machine. To detect five distinct plant leaf illnesses, diseased patches were segmented by clustering the preprocessing-steps-obtained properties with K-means, and then classified with Artificial Neural Networks (ANN) after color- and texture-based feature extraction.

An experiment was conducted to detect and identify two distinct viruses that manifest on the leaf of the tomato and cause the disease. Using features derived from the Local Binary Pattern approach, the SVM method was able to identify three distinct leaf diseases in grapevines. The application of image processing-based candidate hot-spot identification and Naive Bayes classifier for mobile-based early diagnosis of three distinct wheat illnesses was carried out, and their proposed solution was deployed on cellphones and assessed in a real-world setting. Conventional crop disease scouting still mostly depends on visual inspection of leaf colour patterns and crown architecture. Observing disease symptoms on plant leaves with the naked eye and diagnosing plant diseases based on experience is time-consuming, labor-intensive, and requires specialised knowledge. Due to the diversity of plants, the disease features of different crops are also distinct; this fact adds a great deal of complexity to the classification of plant diseases. In the meanwhile, numerous studies have focused on classifying plant diseases using machine learning.

2. LITERATURE REVIEW:

Tomato crop disease classification using pre-trained deep learning algorithm:

- Aravind et al. in this paper describes the disease classification in tomato crop using deep learning techniques. The proposed methodology classifies image into 7 classes with 6 diseases and 1 healthy class. For analysis, AlexNet and VGG16 have been incorporated. The dataset was taken from PlantVillage Dataset. The dataset consisted of 13,262 segmented images. Transfer Learning approach was used. The accuracy obtained using AlexNet and VGG16 net are 97.49% and 97.23% respectively. Hyper parameter tuning was done for analyzing the performance with different parameters and maximum accuracy was found when the number of images is 373.

Plant disease classification using deep learning: -

(Akshai KP & J. Anitha, 2021) in this paper have proposed a deep learning approach for the classification of 4062 images into 4 different classes. The images considered are of Grape plant leaves. The classed are divided as follows: (1) 423 healthy leaves (2) 1382 Esca Affected (3) 1180 black rot affected leaves (4) 1076 Leaf Blight affected images. The proposed method includes deploying four different models viz. CNN model, VGG model, RES-NET and DENSE-NET. The accuracies achieved for each model is 94.58%, 95.32%, 97.04%, 98.27% respectively.

Using Deep Learning for Image-Based Plant Disease Detection: -

Sharada et al. in this paper discusses image-based plant disease detection using deep learning architectures viz. AlexNet and GoogleNet. The experiment was carried out on a publicly available dataset of 54,306 images of diseased and healthy plant leaves collected under controlled conditions and using deep learning, 14 crop species are identified and 26 diseases are classified. The dataset has 38 class labels. The trained model gave 99.35% accuracy. To achieve maximum accuracy, trial and error approach is used for train – test split. For comparison of overall different experimental configurations, F1 core is used. The main differentiator in this method is the absence of any feature engineering for classification. Limitations faced by the model are reduced accuracy under different atmospheric conditions and constraint of using single leaf for classification in homogeneous background.

Plant Disease Classification Using Deep Learning Methods: -

(Hu Wan et al. 2020) proposed several deep learning models particularly transfer learning using pre-trained image weights. The dataset consists of 36258 images of 10 different types of plants and classifying these images into 59 different types of leaves with different diseases. The models proposed are AlexNet, ResNet-34, ResNet-50, VGG-11, VGG-19, Inception-V3 and accuracies of 81.2%, 87.81%, 87.73%, 86.79%, 84.55%, 87.99% respectively

3. PROPOSED WORK:

3.1 Data Set

In this study, PlantVillage dataset is used containing 38 classes and 54305 images of 14 different plant species in total, 12 of which are healthy, 26 of which are diseased. Images in the dataset are colored images of varying sizes. The dataset also has one more class identifying 1143 background images. Some data was obtained from the ground too containing apple cedar images. Thus, the total number of images in the dataset is 55448.

	Number
Tomato___Late_blight	1851
Tomato___healthy	1926
Grape___healthy	1692
Orange___Haunglongbing_(Citrus_greening)	2010
Soybean___healthy	2022
Squash___Powdery_mildew	1736
Potato___healthy	1824
Corn_(maize)___Northern_Leaf_Blight	1908
Tomato___Early_blight	1920
Tomato___Septoria_leaf_spot	1745
Corn_(maize)___Cercospora_leaf_spot Gray_leaf_spot	1642
Strawberry___Leaf_scorch	1774
Peach___healthy	1728
Apple___Apple_scab	2016
Tomato___Tomato_Yellow_Leaf_Curl_Virus	1961
Tomato___Bacterial_spot	1702
Apple___Black_rot	1987
Blueberry___healthy	1816
Cherry_(including_sour)___Powdery_mildew	1683
Peach___Bacterial_spot	1838
Apple___Cedar_apple_rust	1760
Tomato___Target_Spot	1827
Pepper,_bell___healthy	1988
Grape___Leaf_blight_(Isariopsis_Leaf_Spot)	1722
Potato___Late_blight	1939
Tomato___Tomato_mosaic_virus	1790
Strawberry___healthy	1824
Apple___healthy	2008
Grape___Black_rot	1888
Potato___Early_blight	1939
Cherry_(including_sour)___healthy	1826
Corn_(maize)___Common_rust_	1907
Grape___Esca_(Black_Measles)	1920
Raspberry___healthy	1781
Tomato___Leaf_Mold	1882
Tomato___Spider_mites Two-spotted_spider_mite	1741
Pepper,_bell___Bacterial_spot	1913
Corn_(maize)___healthy	1859

Fig.1 Overview of the Dataset

Fig. 2 Different leaf image having diseases

3.2 Transfer Learning:

Transfer learning is a machine learning technique in which the knowledge gained during training in a problem is used for training in another task or field (Weiss et al., 2016). In deep learning, the first few layers are trained to define the characteristics of the task. During transfer learning, the last few layers of the trained network can be removed and retrained with new layers for the target task. In the transfer learning approach, using the knowledge of the network previously trained with large amounts of visual data in a new task is very advantageous in terms of saving time and achieving high accuracy compared to training a model from scratch.

4. IMPLEMENTATION DETAILS:

4.1) Data Visualization and Exploration

The data used is quite big and is divided into many input classes. Below given figure 3 shows how the data is divided into various classes and the number of samples in each class. From the figures, it is evident that Tomato crop holds the highest number of samples for observation and has the greatest number of output classes viz. 10, while the squash soybean constitutes the least number and has only 1 output class. Thus, from the data, we can see that the dataset is widely spread and needs to be analyzed and processed very carefully.

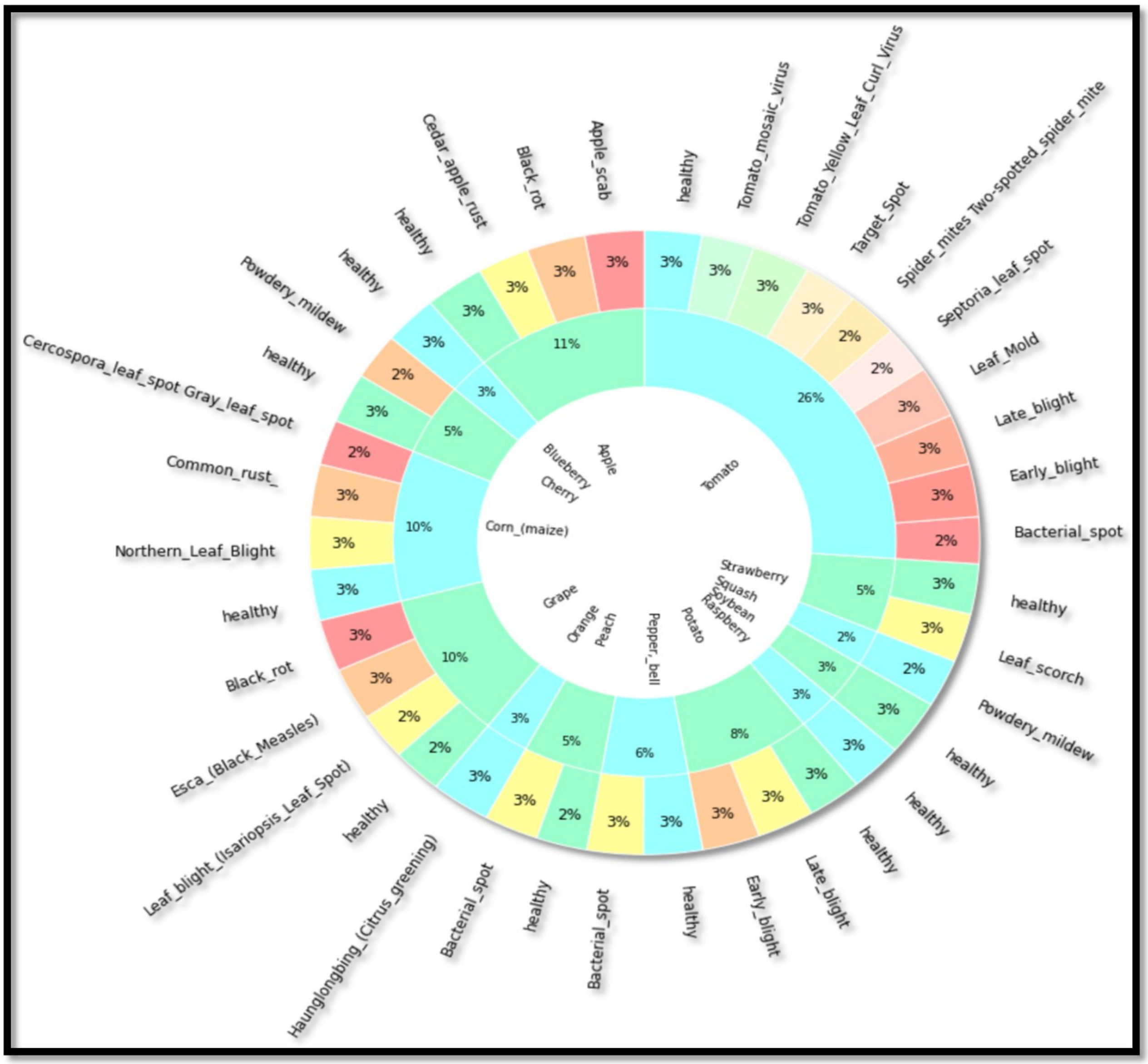

Fig. 3 Data variety

Figure 4 and figure 5 shows the data distribution in each class for both training and validation sample set

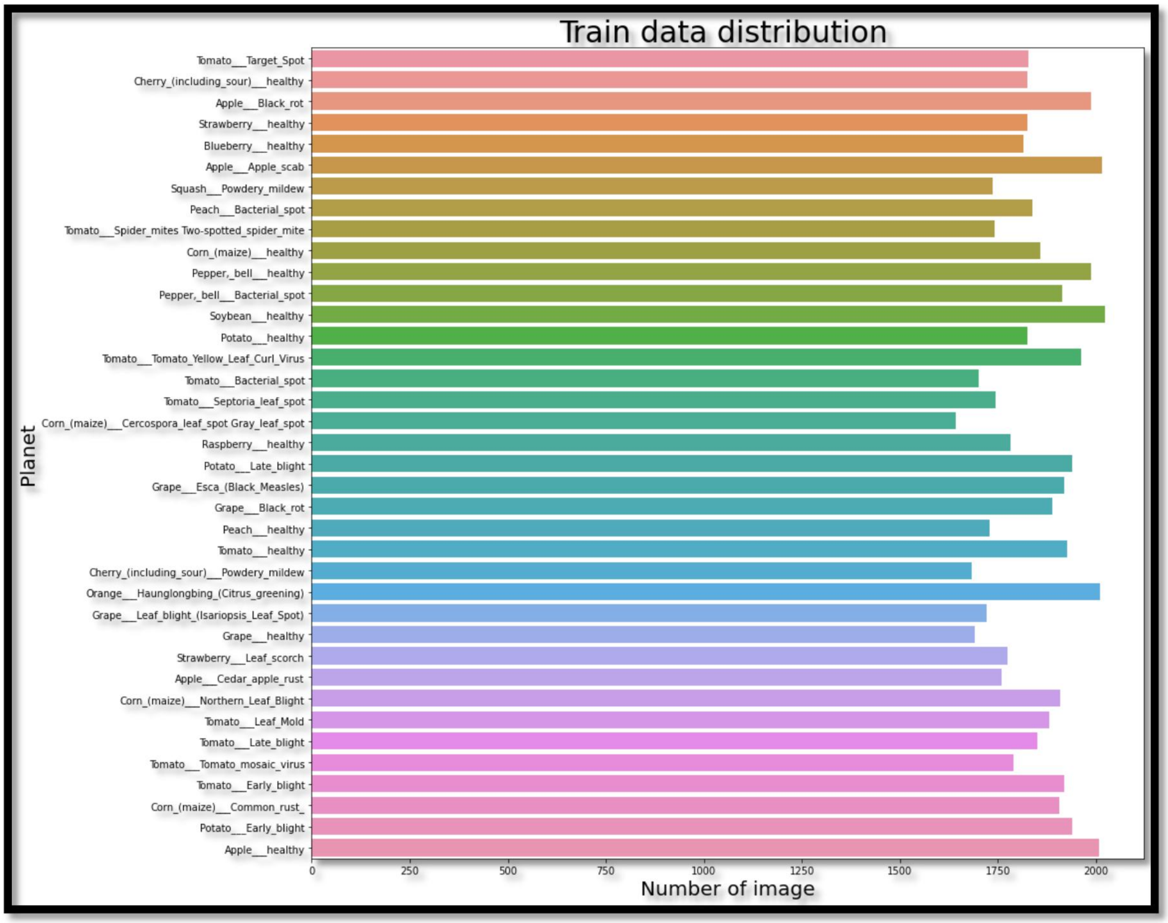

Fig.4 Number of samples in Training sample set

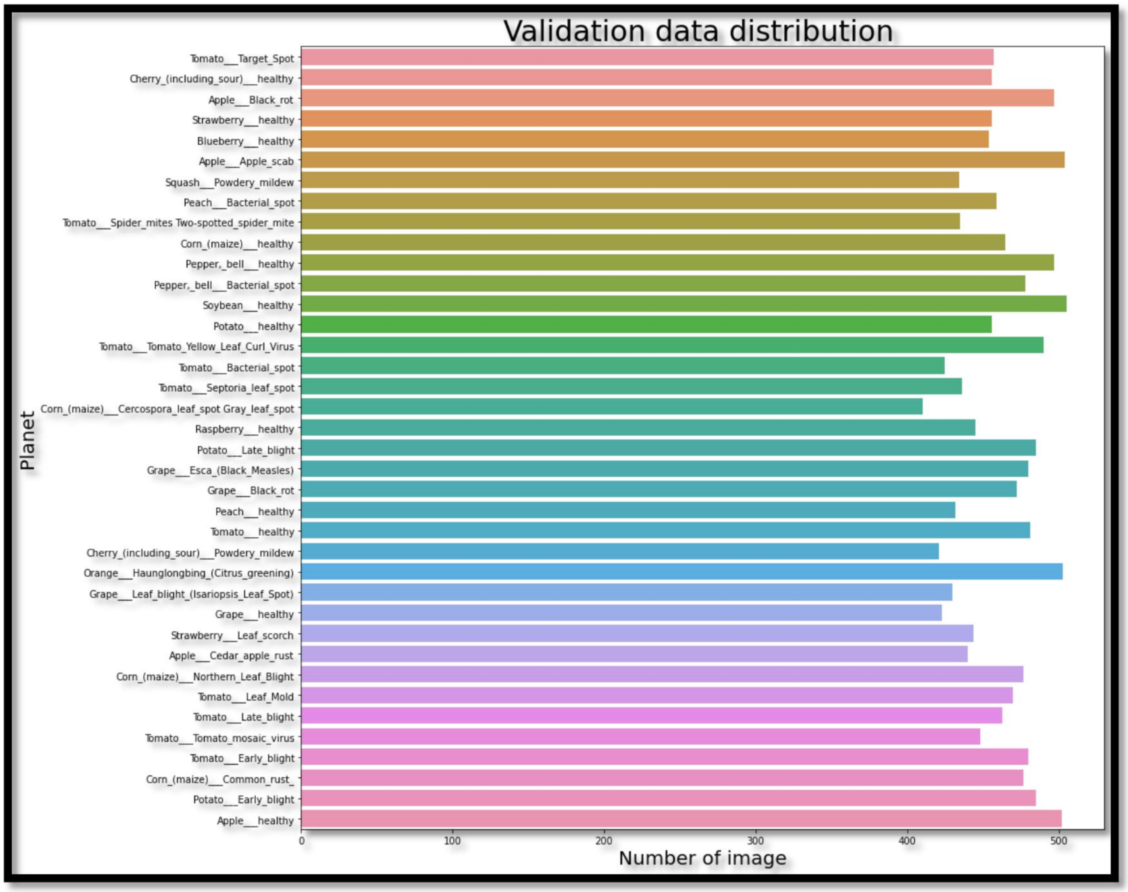

Fig. 5 Number of images in Validation Sample Set

4.2) Feature Extraction

Deep Learning reduces the tedious process of Feature Selection and automates it and thereby, reduces the chances of error. Deep Learning improves the computational efficiency so is a right choice to proceed with it. CNN proves to be effective in the classification of images by extracting important features.

4.3) MobileNet V2

MobileNetv2 by Google is an inverted residual structure where non-linearities in narrow layers are removed. In MobilenetV2, feature extraction is focused majorly. The inverted structure preserves the information and solves the problem of information getting left out in non-linear layers. The structure of MobileNetV2 is as below.

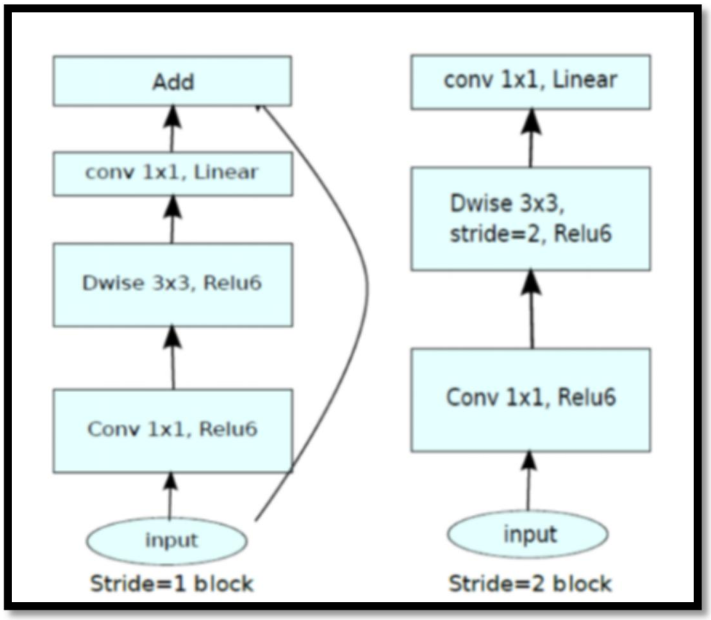

Fig. 6 MobileNetV2 architecture used in the model

In mobilenetV2, there are 2 types of blocks, residual block with stride of 1 and other with stride of 2 to downsize the image. Both the blocks have 3 layers. The first layer is 1 X 1 convolutional layer with the ReLU activation function, the second layer is depthwise convolution, and the third layer is 1 X 1 convolution without any non-linearity. Table 1 shows the layer description of MobileNetV2 architecture.

Table 1. A typical MobileNetV2 layer

Input	**Operator**	**Expansion factor**	**Output channel**	**Repeating number**	**stride**
224*224*3	Conv2d	-	32	1	1
112*112*32	Bottleneck	1	12	1	1
112*112*16	Bottleneck	6	24	2	2
56*56*24	Bottleneck	6	32	3	2
28*28*32	Bottleneck	6	64	4	2
14*14*64	Bottleneck	6	96	3	1
14*14*96	Bottleneck	6	160	3	2
7*7*160	Bottleneck	6	320	1	1
7*7*320	Conv2d 1X1	-	1280	1	1
7*7*1280	Avgpool 7X7	-	-	1	-
1*1*1280	Conv2d 1X1	-	K	-	-

MobileNetV2 has a very low parameter count than the original MobilentV1 and supports any input size greater than 32 X 32 with larger image size giving better performance. Our MobileNetV2 consists of 4116070 as shown in the figure 7, which is way more than the original model having around 3 million total parameters.

```
Total params: 4,116,070
Trainable params: 4,080,934
Non-trainable params: 35,136
```

Fig. 7 Model Parameters

4.4) Resnet-50

For Deep Learning, it is hard to train because of the vanishing gradient problem. Residual Network is a solution to this problem. ResNet uses the concept of skip connection. ResNet 50 means that the model has 50 layers. ResNet 50 brings more detection accuracy. In ResNet, the convolutional layers are stacked one after the other like traditional CNN but the main difference is the availability of original input to the output blocks. This helps in maintaining the values of output after every block so that it does not diminish during backpropagation. Below given figure 2 is the ResNet 50 architecture used in our proposed work. Figure 8 shows the overall block diagram with output classifier.

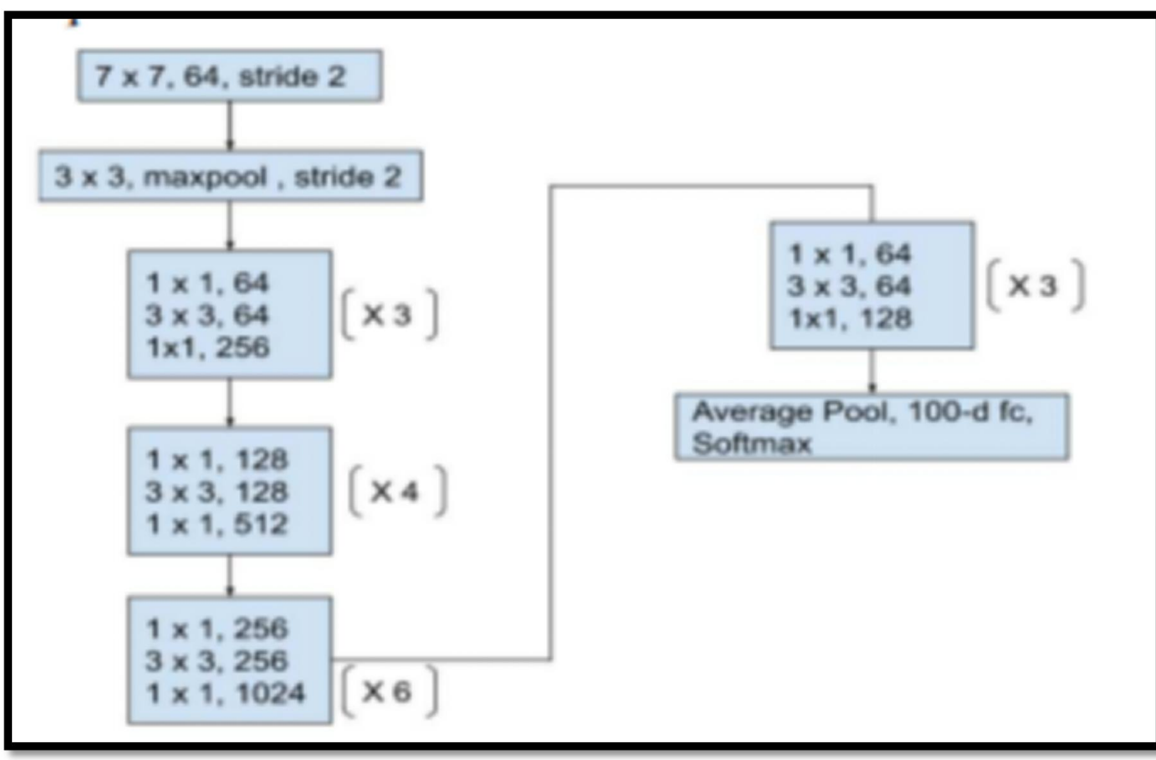

Fig. 8 RESNET – 50 architecture

5) Result analysis

After running for 10 epochs we have got the testing accuracy as 97% and training accuracy as 98.08%. It is shown in the below given figure 9.

```
Epoch 1/10
3514/3514 [==============================] - 1167s 328ms/step - loss: 0.4183 - accuracy: 0.8724 - val_loss: 2.1101 - val_accuracy: 0.6295
Epoch 2/10
3514/3514 [==============================] - 1113s 317ms/step - loss: 0.2225 - accuracy: 0.9312 - val_loss: 1.1074 - val_accuracy: 0.7829
Epoch 3/10
3514/3514 [==============================] - 1142s 325ms/step - loss: 0.1635 - accuracy: 0.9488 - val_loss: 0.8080 - val_accuracy: 0.8234
Epoch 4/10
3514/3514 [==============================] - 1172s 333ms/step - loss: 0.1387 - accuracy: 0.9557 - val_loss: 1.1895 - val_accuracy: 0.7721
Epoch 5/10
3514/3514 [==============================] - 1165s 331ms/step - loss: 0.1093 - accuracy: 0.9649 - val_loss: 0.2382 - val_accuracy: 0.9365
Epoch 6/10
3514/3514 [==============================] - 1178s 335ms/step - loss: 0.0941 - accuracy: 0.9703 - val_loss: 0.6003 - val_accuracy: 0.8670
Epoch 7/10
3514/3514 [==============================] - 1157s 329ms/step - loss: 0.0880 - accuracy: 0.9723 - val_loss: 0.2161 - val_accuracy: 0.9405
Epoch 8/10
3514/3514 [==============================] - 1154s 328ms/step - loss: 0.0784 - accuracy: 0.9753 - val_loss: 0.6923 - val_accuracy: 0.8218
Epoch 9/10
3514/3514 [==============================] - 1121s 319ms/step - loss: 0.0717 - accuracy: 0.9766 - val_loss: 1.3563 - val_accuracy: 0.7725
Epoch 10/10
3514/3514 [==============================] - 1120s 319ms/step - loss: 0.0608 - accuracy: 0.9808 - val_loss: 0.5811 - val_accuracy: 0.8548
```

Fig. 9 Epochs Running with Validation Accuracy

Fig. 10 shows the classification report of the MobileNetV2 model used. Figure 11 and 12 shows the training and testing accuracy. Figure 13 shows the Confusion matrix of the dataset.

	precision	recall	f1-score	support
0	0.93	0.98	0.95	504
1	0.88	1.00	0.94	497
2	1.00	0.89	0.94	440
3	0.69	0.98	0.81	502
4	0.67	0.98	0.80	454
5	0.98	0.93	0.96	421
6	0.96	1.00	0.98	456
7	0.95	0.93	0.94	410
8	1.00	0.98	0.99	477
9	0.94	0.97	0.95	477
10	1.00	0.99	1.00	465
11	0.97	0.99	0.98	472
12	0.97	0.99	0.98	480
13	1.00	0.83	0.91	430
14	0.97	1.00	0.98	423
15	1.00	0.66	0.79	503
16	0.84	0.96	0.90	459
17	1.00	0.41	0.59	432
18	0.84	1.00	0.91	478
19	0.78	1.00	0.88	497
20	0.97	0.99	0.98	485
21	1.00	0.87	0.93	485
22	0.91	0.93	0.92	456
23	1.00	0.34	0.50	445
24	0.95	0.96	0.96	505
25	0.99	1.00	1.00	434
26	1.00	0.95	0.97	444
27	0.62	0.99	0.76	456
28	0.99	0.56	0.72	425
29	0.97	0.58	0.73	480
30	0.87	0.84	0.86	463
31	0.48	0.99	0.65	470
32	0.42	0.77	0.55	436
33	1.00	0.32	0.48	435
34	0.63	0.83	0.72	457
35	0.97	0.68	0.80	490
36	0.93	0.94	0.94	448
37	1.00	0.26	0.41	481
accuracy			0.85	17572
macro avg	0.90	0.85	0.84	17572
weighted avg	0.90	0.85	0.84	17572

Fig. 10 Classification Report

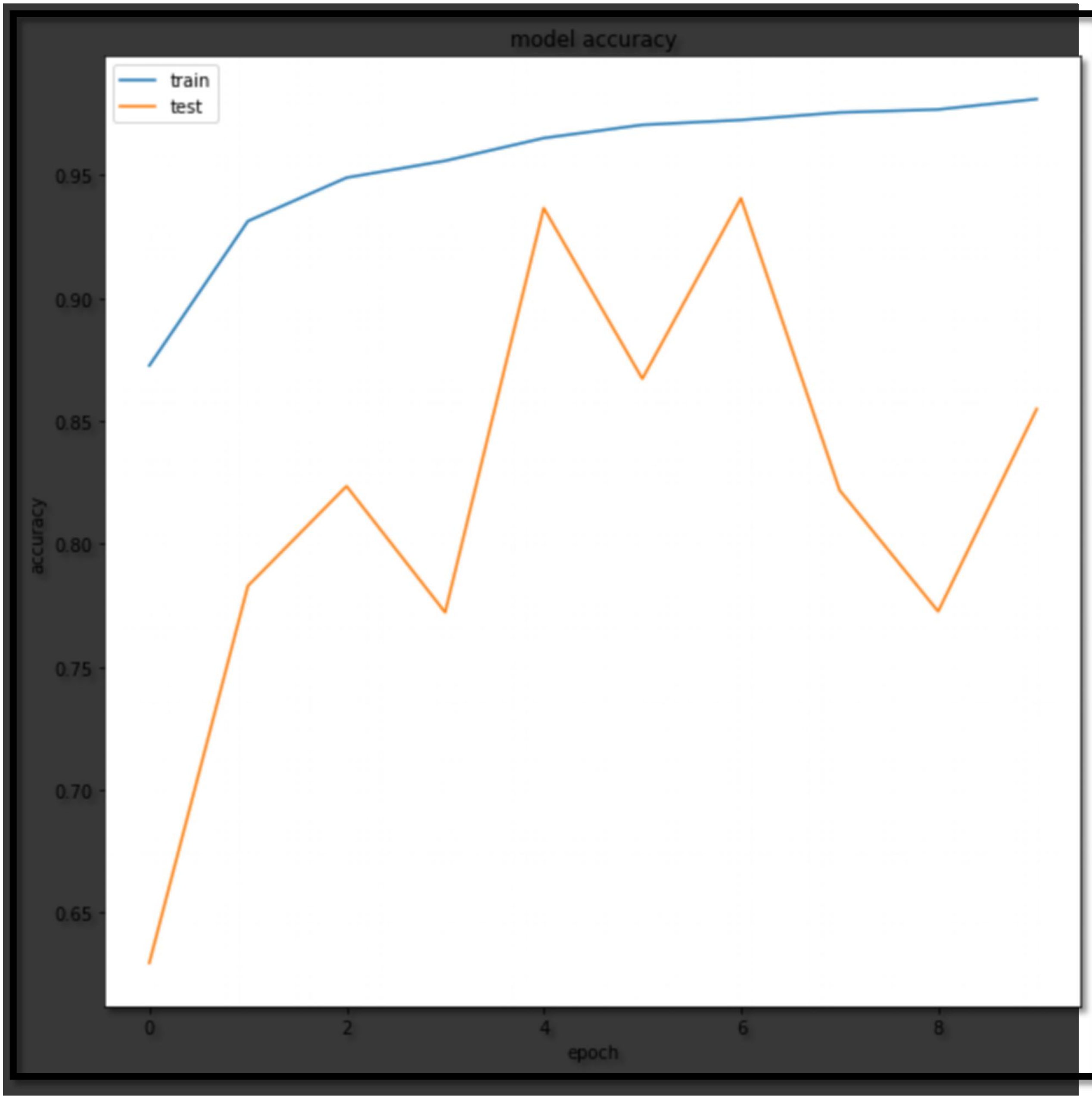

Fig. 11 Training and testing Accuracy

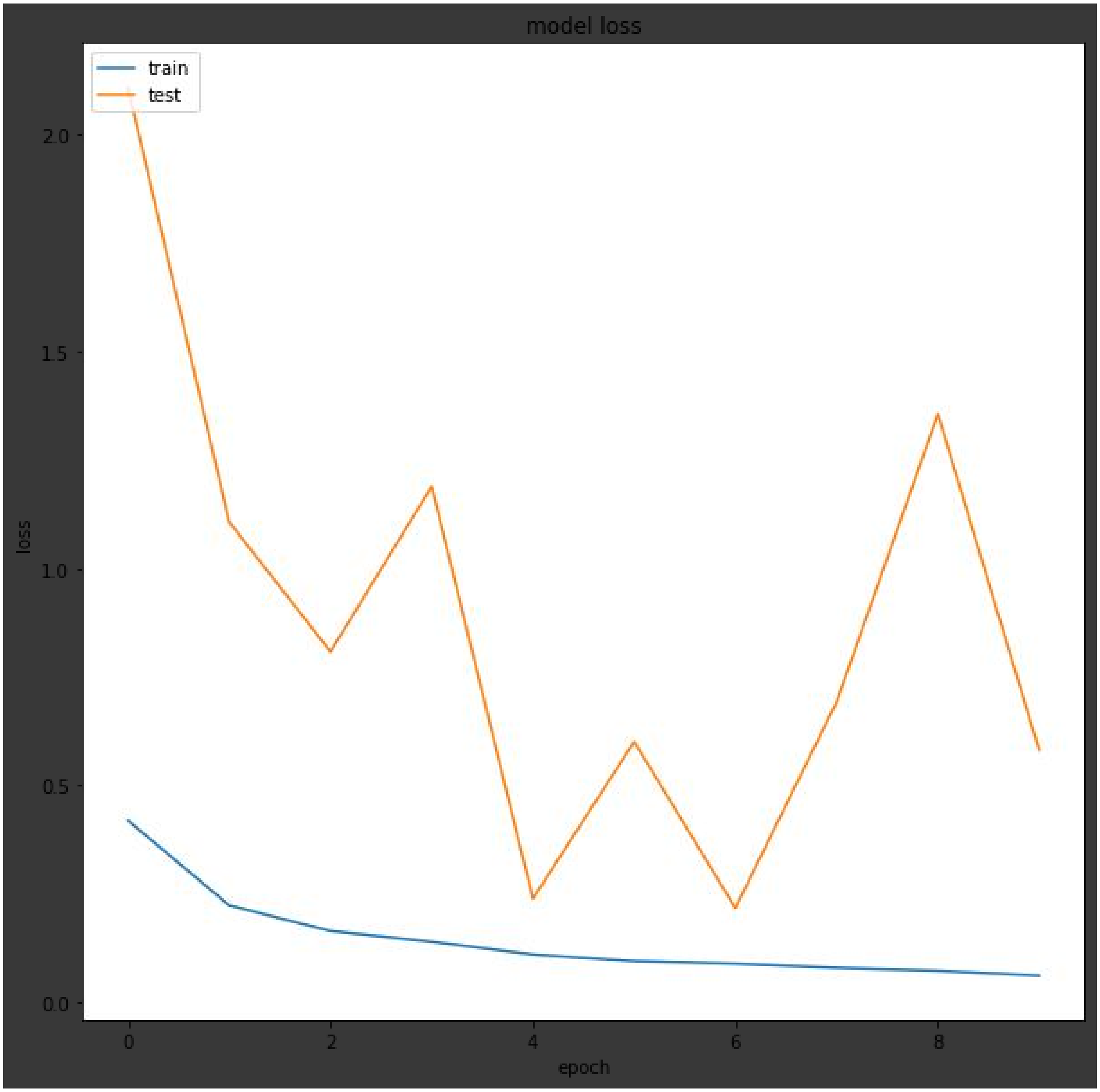

Fig. 12 Train and test loss

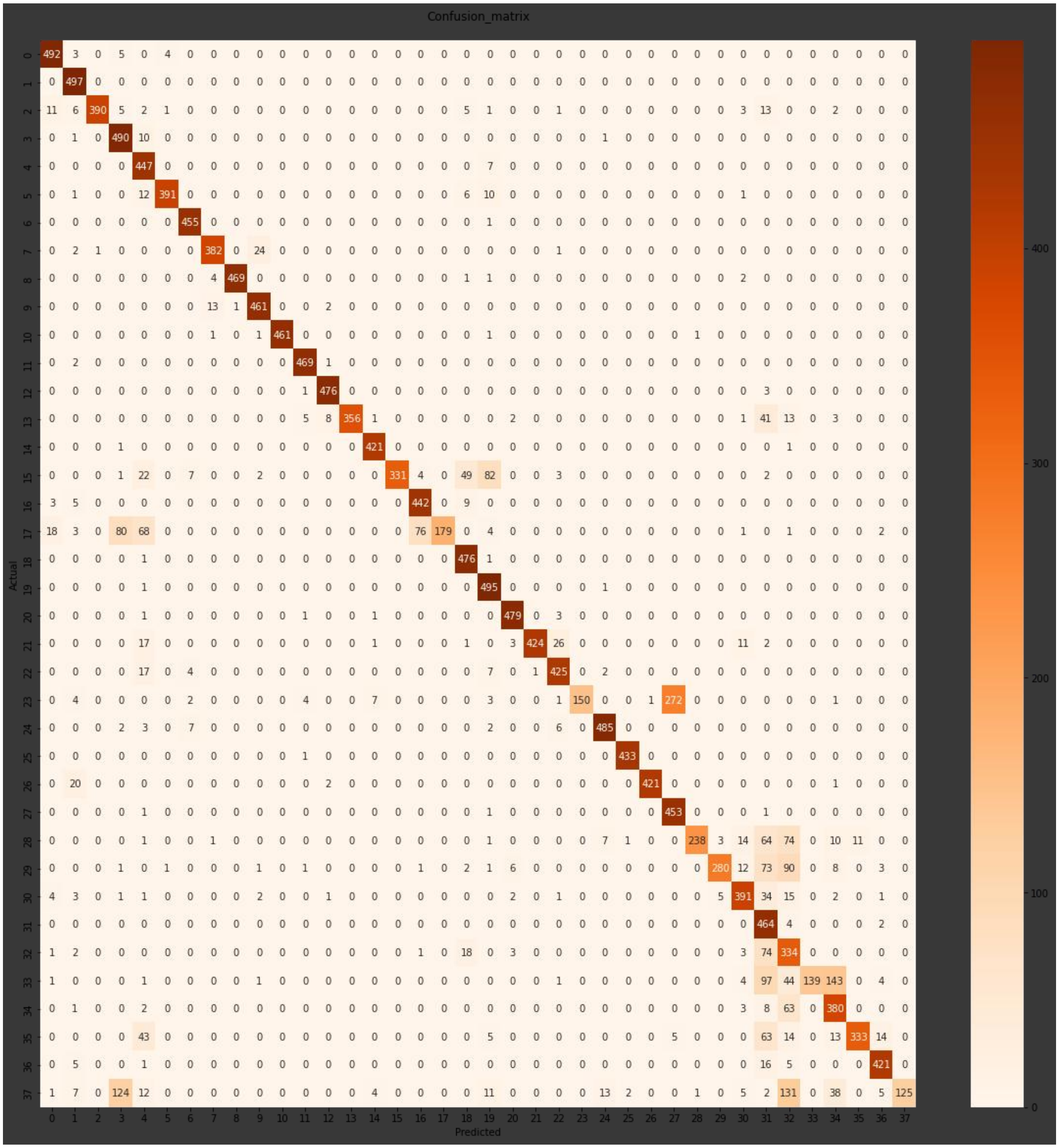

Fig. 13 Confusion Matrix

Similarly, following the steps for RESNET - 50, an accuracy of 94.34% was obtained.

6. Conclusion: -

From the above experimentation, it is concluded that newly developed model MobileNetV2 which is an updated version of MobilenetV1, is a better model for implementation as compared to a more commonly used CNN transfer learning model Resnet or AlexNet, VGGNet etc. The MobileNetV2 achieved testing accuracy of 97% (due to its unique architecture), which is comparatively more than the standard models.

7. References: -

1. https://www.sciencedirect.com/science/article/pii/S1877050918310159

2. DOI: - 10.1109/ICSPC51351.2021.9451696

3. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5032846/

4. https://dl.acm.org/doi/10.1145/3380688.3380697

5. https://arxiv.org/abs/1801.04381

6. https://towardsdatascience.com/review-mobilenetv2-light-weight-model-image-classification-8febb490e61c

7. https://towardsdatascience.com/understanding-and-coding-a-resnet-in-keras-446d7ff84d33