International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 05 | May 2019 www.irjet.net p-ISSN: 2395-0072

CROP MONITORING: Using MobileNet Models

Vinutha M S[1], Riya Kharbanda[2], Rashmi B[3], Rajani S N[4], Rajesh Kumar Pareek[5]

[1]Assistant Professor, Dr Ambedkar Institute of Technology, Bengaluru
[2]Student, Dr Ambedkar Institute of Technology, Bengaluru
[3]Student, Dr Ambedkar Institute of Technology, Bengaluru
[4]Student, Dr Ambedkar Institute of Technology, Bengaluru
[5]Student, Dr Ambedkar Institute of Technology, Bengaluru

Abstract - *Agriculture plays a vital role in India's economy. Over 58 percent of the rural households depend on agriculture as their principal means of livelihood. However, the farmers of India have been facing a lot of real time challenges like crop diseases. They are major threat to food security, but their immediate identification remains difficult in many places due to the lack of proper facilities. However, increasing smartphone penetration along with advancement in computer vision that are made possible via deep learning, have paved the way for smartphone-assisted disease diagnosis. Using a public dataset of 16,471 images of diseased and healthy plant leaves collected under controlled conditions, we train a deep convolutional neural network to identify 8 crop species and 15 labelled diseases. The trained model achieves an accuracy of 99% on the testing dataset.*

Key Words*: *Image Processing, Deep Learning, Android Application*, *Tensorflow.

1. INTRODUCTION

Agriculture is an important source of income for Indian people. Farmers can grow variety of crops but diseases hamper the growth of crops. One of the major factors responsible for the crop destruction is plant disease. Different plants suffer from different diseases. The main part of plant to examine the disease is leaf. The major categories of plant leaf diseases are based on viral, fungal and bacterial infections. The diseases on leaf can reduce both the quality and quantity of crops and their further growth. The easy method to detect the plant diseases is with the help of agricultural expert having knowledge of plant diseases. But this manual detection of plant diseases takes lot of time and is a tedious work.

To overcome these challenges a mobile application was developed using a deep neural network that is trained on 16,471 images of 8 different classes of leaves. The image dataset is using Kaggle's PlantVillage database.The application was made to do the binary class categorization on the input images that are fed to the application via a mobile camera in real time.

This paper describes an efficient network architecture in order to build very small, low latency models that can be easily matched to the design requirements for mobile and embedded vision applications. Section 3 reviews prior work in building small models. Once the model is trained and tested it can be deployed for use on the android platform. Training and testing of the model is done in a Linux based system while Android development is using a windows operating system.

Section 4 describes the MobileNet architecture that defines smaller and more efficient MobileNets . Section 5 illustrates implementation where the application is optimized to work on mobile devices by normalizing the weights and freezing some variables. This process of optimization makes the graph file 4 times lighter as compared to a traditional neural network architecture. Section 6 gives the results for the experiment carried out. Section 7 closes with a conclusion.

2. KNOWING CROP DISEASES

The Table-1 below gives details of the sample dataset having crops which are used in training process of the model.

Table -1: Few examples of crop diseases.

DISEASE	DESCRIPTION
Apple Scab	Caused By: Fungal infection Symptoms: Leaves become twisted, form olive green spots on the upper surface. Remedy : Spray liquid copper soap,Bonide plant fungicide, Organocide.
Peach Bacterial Spot	Caused By: "bacterium xanthomonas campestris pvpruni" Symptoms: Purple-brown spots on foliage followed by centre of lesion falling out.

	Remedy : Chemical spray with copper based bactericide and the antibiotic oxytetracycline.
Corn Common Rust	Caused By: Fungal disease by "puccinia sorghi", favoured by moist. Symptoms: Lesions on leaves, poorly filled kernels. Remedy : Apply a foliar fungicide like Headline AMP, PropiMax, Quadris, Quil.
Grape Black Measles	Caused By: Fungal disease by "phaeomoniella chlamydospora" Symptoms: Form tiger stripe pattern on the upper leaf surface. Remedy : Lime sulfur sprays help control the disease.
Orange Huanglongbing	Caused By: Insect known as "Diaphorina citri" Symptoms: Yellowing of the leaves. Remedy : Make use of foliar pyrethroid insecticide to quickly kill adults and immature psyllids.
Potato Early Blight	Caused By: Fungal pathogen called "Alternaria solani" Symptoms: Circular dark brown spots on the leaves. Remedy : Apply a copper based fungicide/foliar spray, Organocide Plant Doctor.
Strawberry Leaf Scorch	Caused By: Fungus known as "Diplocarpon earliana" Symptoms: Small purplish blemishes that occur on the topside of leaves. Remedy : Apply sulfur sprays or copper-based fungicides weekly at first sign of disease to prevent its spread.
Tomato Spidermites	Caused By: Attack of insects known as "Tetranychidae" Symptoms: Leaves show patterns of tiny spots or stipplings. Remedy : Keep the humidity high, Discard or burn the insect invaded leaves.

3. EXISTING WORK

The existing methodology for disease detection is just an optic observation by specialists through which identification and detection of plant diseases is completed. Moreover, an oversized team of specialists still are needed for continuous monitoring, that inturn prices terribly high, once farms are massive. At an equivalent time, in some countries, farmers don't have correct facilities. In such condition, the advised technique proves to be helpful in watching massive fields of crops. An automatic detection of the diseases by simply seeing the symptoms on the plant leaves makes it easier still as cheaper. Until now, several technologies have emerged, that proved useful in disease detection without any specialist's need. However, there's no application that could assist farmers without having much technical knowledge to use such technologies.

In traditional crop disease detection methods, convolutional neural network uses multilayer convolution to extract features and combine them automatically. It also uses the pooling layer, fully connected layer and softmax. Google made TensorFlow open source that is used for arithmetic calculation, specializing in machine learning applications. Second generation of Google artificial intelligence learning system got much attention and affirmation in the field of machine learning all over the world. TensorFlow has advantages of high accessibility, high flexibility, and high provision of TensorFlow researchers through github and online forum to progress its efficiency. Today, Google has unconfined number of pretrained models on the TensorFlow's official website[10], to expedite the use of researchers in different fields. MobileNets is one of the pretrained models on the TensorFlow. It is a continuous improvement to the initial structure of computer vision after Inception-v1, Inception-v2, Inception-v3 in 2015.

The MobileNet model is trained on the ImageNet datasets, comprising the facts that can identify 1000s of categories in ImageNet, the fault percentage of top-5 is upto 3.5%, the fault percentage of top-1 dropped to 17.3%.Transfer learning is a new machine learning method which can use the existing knowledge learned from one environment and solve the new problem which is different but has some relation with the old one. TensorFlow provides comprehensive tutorials to reskill commencement's final layer for new classifications by means of transfer learning. Compared with the traditional neural network, it only needs to use a small quantity of data to train the model, and attain high exactitude with a short training time.

Depth wise separable convolutions have been shown to be successful model used for image classification, in both the cases, obtaining better models than previously possible for an available parameter count and significantly dropping the number of parameters essential to perform at a given level.

4. NEURAL NETWORK ARCHITECTURE

In this section, we describe the core architecture layer of MobileNets[6] which is built on depth wise distinguishable filters . In MobileNets the depth wise intricacy is used and that applies a distinct filter to each response channel. The MobileNet model is constructed on depth wise distinguishable intricacies which happens to be a form of factorized complexities that factorizes a standard complexity into a depth wise complexity and a 1x1 complexity is termed as pointwise complexity. After that the point wise complexity applies a 1x1 complexity to add the outputs to the depth wise complexity.

In a standard complexity, both are filtered and add inputs to form a different output set in one step. The depthwise distinguishable complexity separates this into two layers, one for filtering and the other for linking. By doing this separation it has a huge effect of reduction with computational time and size of the model. Figure-1 indicates diverse scenes of how a regular complexity 1(a) is factorized into a depthwise complexity1(b) and a 1x1 pointwise complexity 1(c). The first layer is a fully connected layer. Depthwise convolution is used for applying a single filter on every input channel while pointwise convolution is used to form a linear combination of the output from the depthwise layer.

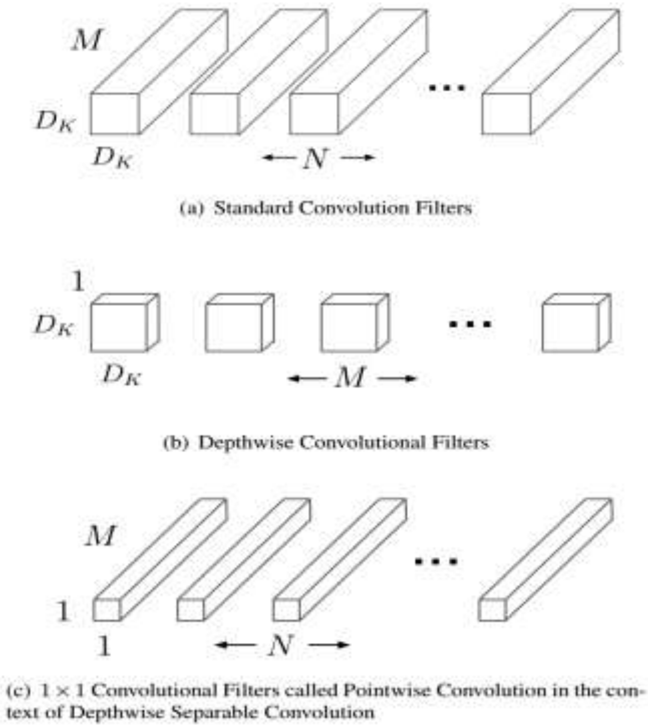

Figure-1 Standard convolutional filters in (a) are replaced by two layers as shown in (b) and (c).[6]

There are two non-linearities used: batchnorm and ReLU after each layer as shown in Figure-2.

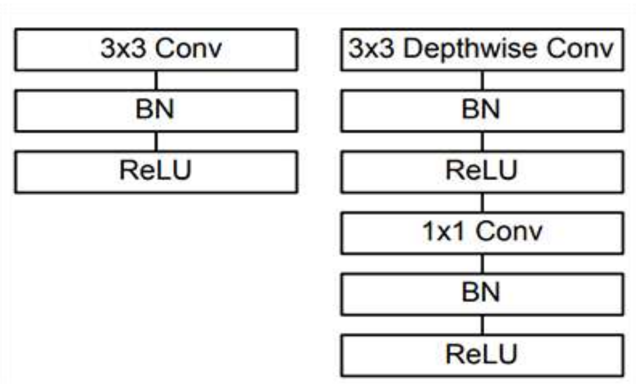

Figure-2 Left:Standard convolutional layer, Right-Depthwise Convolutional layer.[6]

5. IMPLEMENTATION

This section focuses on experimental setup for crop disease model using MobileNet on TensorFlow framework. Here classification model is separated into following 3 stages: image preprocessing, training, android application.

5.1 Image Dataset and Preprocessing

In the image preprocessing step we need to label the data, as shown in Figure-3, since the learning method of convolution neural network fits into administered learning in machine learning.

Figure-3 A glimpse of crop disease dataset.

5.2 The Training Process.

The MobileNets training is done in Tensorflow with the help of asynchronous gradient descent having 4000 training steps. The layer architecture of the MobileNets model is given in Table-2. In comparison to other models such as Inception, the MobileNets use less regularization and data augmentation. In fact, the size of the input to the network is also small.

The output of the neural network is 15 class labels of 8 different crops. The architecture is trained and tested using Python language with Tensorflow CPU library installed. Figure-4 shows the MobileNets architecture generated from tensorboard that is an inbuilt feature of tensorflow library in Python.

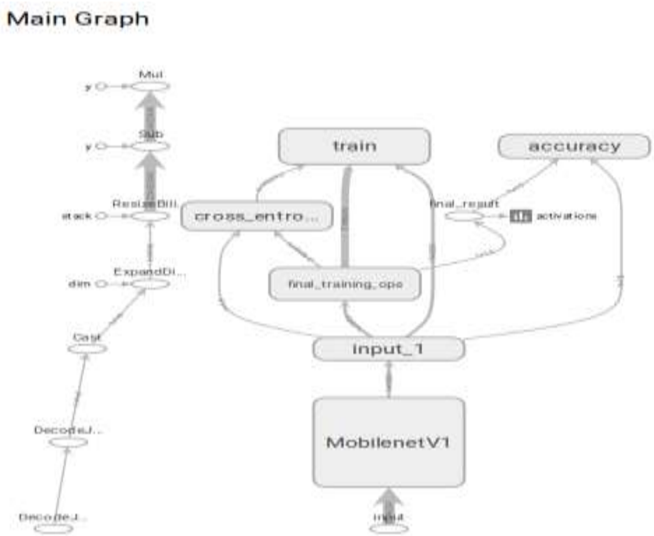

Table-2 MobileNet Architecture[6]

Type/Stride	Filter Shape	Input Size
Conv/s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Conv dw/ s1	$3 \times 3 \times 3 \times 32\ dw$	$112 \times 112 \times 32$
Conv/s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Conv dw/s2	$3 \times 3 \times 64\ dw$	$112 \times 112 \times 64$
Conv/s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Conv dw/s1	$3 \times 3 \times 128\ dw$	$56 \times 56 \times 128$
Conv/s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Conv dw/s2	$3 \times 3 \times 128\ dw$	$56 \times 56 \times 128$
Conv/s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Conv dw/s1	$3 \times 3 \times 256\ dw$	$28 \times 28 \times 256$
Conv/s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Conv dw/s2	$3 \times 3 \times 256\ dw$	$28 \times 28 \times 256$
Conv/s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5 × Conv dw/s1 Conv/s1	$3 \times 3 \times 512\ dw$ $1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$ $14 \times 14 \times 512$
Conv dw/s2	$3 \times 3 \times 512\ dw$	$14 \times 14 \times 512$
Conv/s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Conv dw/s2	$3 \times 3 \times 1024\ dw$	$7 \times 7 \times 1024$
Conv/s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Avg Pool/s1	Pool 7×7	$7 \times 7 \times 1024$
FC/s1	1024×1000	$1 \times 1 \times 1024$
Softmax/s1	Classifier	$1 \times 1 \times 2$

Figure-4 Tensorboard MobileNet control flow model

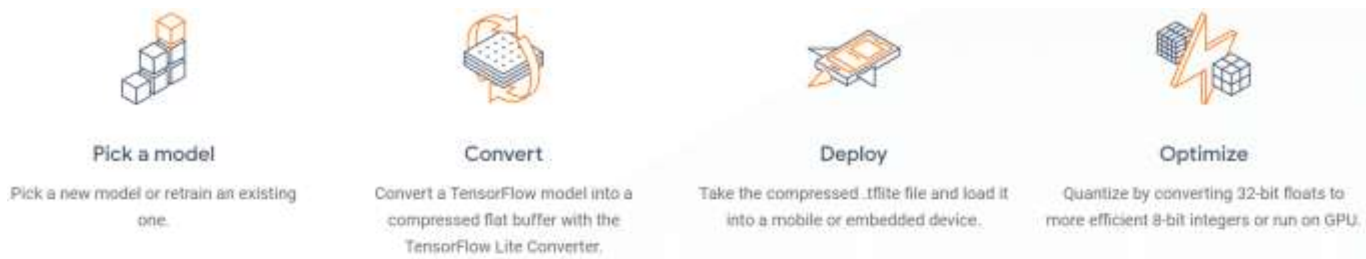

Figure-5 Depicts the procedure for integrating the trained model along with the android application.

5.3 Android Application

After training the neural network model, one gets two files as output: graph file and class labels. Graph file contains all the nodes and operations that are performed during the training of the network. Since the final built application is to be run on a mobile device that has limited capability to perform operation one needs to optimize the graph file before building the application as shown in Figure-5.

The whole neural network is built using tensorflow library that has a built-in tool for removing all the nodes that are not needed for a given set of inputs and outputs. With the help of an optimized model, the number of calculations is reduced by merging the explicit batch normalization. The application is then built using Android Studio, using the TensorFlow Lite dependencies.

6.RESULTS

We would get to see correctly classified crop disease with an accuracy value of between 90% and 99%, though the resulting value will change from case to case since there's arbitrariness in the process of training. On top of that, if we give training for only two crops as two different categories, we should expect higher precision. This number value specifies the percentage of the images in the test set which are provided the exact label after training the model completely.

As it trains, a series of output steps can be seen, each one showing training precision, validation accuracy, and the cross entropy.

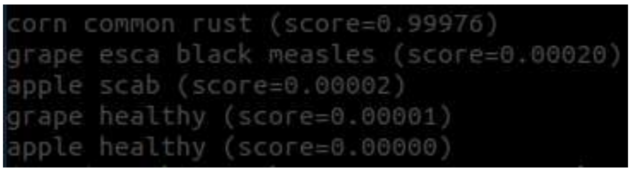

Figure-6 sample result of the MobileNet model

Accuracy of training illustrates the percentage of the images used in the current training batch that were labeled with the precise class. Validation accuracy is the precision; percentage of correctly-labelled images, on a randomly selected images from a different set. Figure-7 shows the accuracy graph that is generated using tensorboard where x and y axis representing the number of training examples and accuracy score.

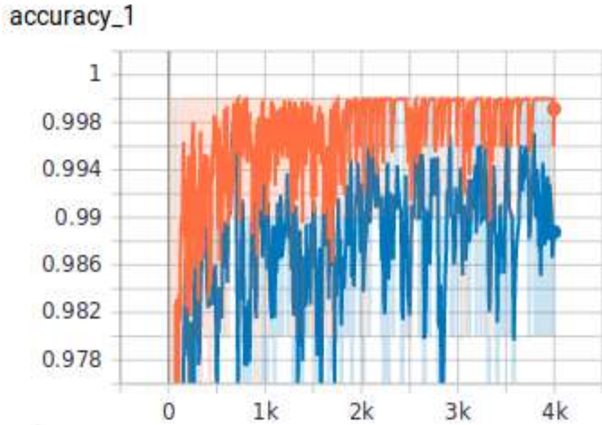

Figure-7 Accuracy w.r.t training(red) and validation(blue)

Cross entropy is a loss function which gives an insight into how good the process of learning is moving ahead, lower numbers are better here. Figure-8 shows the cross entropy graph generated using tensorboard where x and y axis representing the number of training examples and cross entropy score.

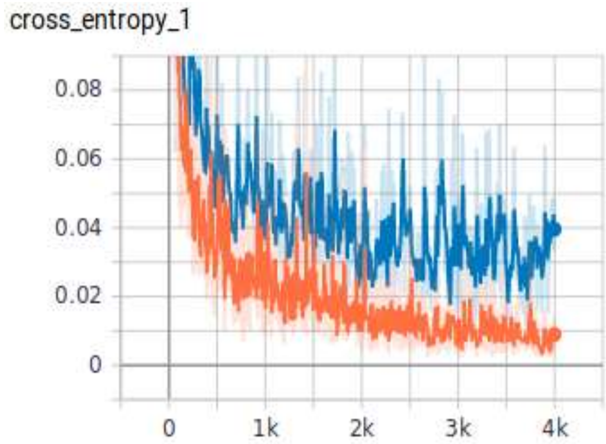

Figure-8 Cross entropy w.r.t training(red) and validation(blue)

If the precision of the training is high but the accuracy of validation remains low the meaning of which is that the network is over fitting, and that it is retaining the specific features in the training images which doesn't help it grouping images more mainly. An exact measure of the networks performance is to quantify its performance on a set of data that is not a subset of training data.

The crop monitoring android application has a simple user interface. The basic idea is to initiate camera triggering, scan the image of the crop, predict the disease type and suggest the corresponding remedy. Figure-9 shows a sample output.

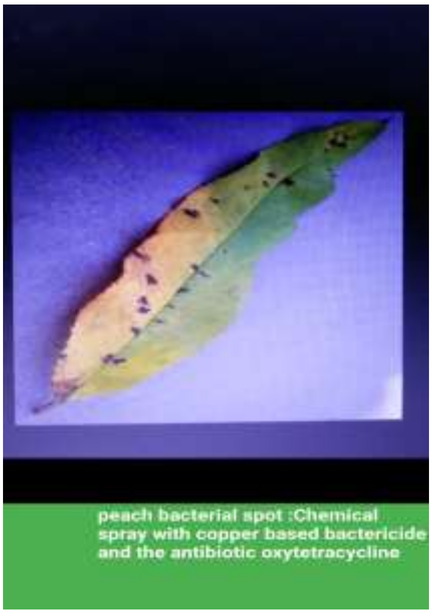

Figure-9 Android app prediction.

7.CONCLUSIONS

We experimented the crop disease detection application using new model architecture called MobileNets based on depthwise separable convolutions. We investigated some of the important design decisions leading to an efficient model. We then built smaller and faster MobileNets using width multiplier and resolution multiplier by trading off a reasonable amount of accuracy to reduce size and latency. We concluded by demonstrating MobileNet's effectiveness when applied to a wide variety of crop image datasets. The MobileNets are optimized to become small and efficient by compromising on the accuracy aspect.

Overall, we studied the existing technologies used for detecting crop diseases, and implemented an efficient way to detect crop diseases where a farmer need not know the details of every disease and can easily detect disease with the help of a mobile app. Also, the app can suggest some pesticides that would be required for curing of the disease.

Moreover, the application can be extended by adding features like location services, text to speech recognition, internet of things based real time triggering, augmented reality, chatbots etc.

REFERENCES

[1] Why MobileNet and Its Variants Are Fast [https://cdn-images-1.medium.com/max/1600/1*XbuW8WuRrAY5pC4t-9DZAQ.jpeg]

[2] CNN Architectures: LeNet, AlexNet, VGG, GoogLeNet, ResNet and more. [https://medium.com/@sidereal/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5]

[3] Deep Learning Model Compression for Image Analysis: Methods and Architectures [https://medium.com/zylapp/deep-learning-model-compression-for-image-analysis-methods-and-architectures-398f82b0c06f]

[4] Review: MobileNetV1—Depthwise Separable Convolution (Light Weight Model)

[5] Nitin R. Gavai, Yashashree A. Jakhade, Seema A. Tribhuvan, and Rashmi Bhattad "MobileNets for Flower Classification using TensorFlow", 2017 International Conference on Big Data, IoT and Data Science (BID) Vishwakarma Institute of Technology, Pune, Dec 20-22, 2017.

[6] Andrew G. Howard , Menglong Zhu, Bo Chen, Dmitry Kalenichenko, and Weijun Wang "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications", arXiv:1704.04861v1 [cs.CV] 17 Apr 2017

[7] Shorav Suriyal, Christopher Druzgalski, and Kumar Gautam "Mobile Assisted Diabetic Retinopathy Detection using Deep Neural Network", 2018 GLOBAL MEDICAL ENGINEERING PHYSICS EXCHANGES/PAN AMERICAN HEALTH CARE EXCHANGES (GMEPE / PAHCE) .

[8] Alex Krizhevsky, IlyaSutskever, and GeoffreyE. Hinton " ImageNet Classification with Deep Convolutional Neural Networks"

[9] Viraj A. Gulhane, Maheshkuma R. H. Kolekar, "Diagnosis Of Diseases On Cotton Leaves Using Principal .

[10] Working of TensorFlow and TensorFlow Lite.[https://www.tensorflow.org/lite].

[11] Sharada P Mohanty, David P. Hughes and Marcel Salathe ."Using Deep Learning for Image –Based Plant Disease Detection".