KLE Society's

KLE Technological University

February 26, 2021

Mini Project
On
Intelligent Disease Management in Maize using Deep Learning Techinque

Submitted by

Name	USN
Karnika Bhardwaj	01FE18BCS085
Sakshi Jha	01FE18BCS182
Sanjay Keshwar	01FE18BCS192
Savitri Khyadad	01FE18BCS196

Team Number:B10

Under The Guidance Of
Dr. Meena S M
Ms. Priyadarshini Patil

SCHOOL OF COMPUTER SCIENCE ENGINEERING
HUBLI-580031

Academic year 2020-2021

Abstract

Karnataka is one of the largest producers of maize in India while maize is the third most widely produced cereal in the country. Agricultural production is affected by various diseases and the infection of various pests in the various crops. To improve productivity early detection if pest infestation and diseases in a crop is essential.

Maize is one among the important crops that is affected by various diseases like leaf blight, common rust etc. Manual methods of disease classification in maize are time consuming as well as less efficient. Deep learning model of Artificial Intelligence makes the manual process less faulty and tedious.

The proposed methodology includes automation of the manual process of detecting the diseases or infestation caused on a maize crop by observing the leaves. Our aim to identify the disease or the pest infestation and suggest preventive measures. The classification of diseases and the pest infestation yielded an accuracy of 80

Acknowledgement

This project is the result of hard work, research, dedication and determination. This project would not have been possible without the help and guidance of so many people. Therefore we would like extend our sincere gratitude to them

We take this opportunity to thank Dr.Ashok Shettar, Vice-chancellor,KLE Technological University and to Dr. Prakash Tiwari, Principal, B V Bhoomaraddi College of Engineering and Technology, Hubli.

First of all we owe a deep sense of gratitude to our project guide Dr. Meena. S. M. (HOD) School of Computer Science Engineering for her valuable guidance and constant help and encouragement throughout the process.

We express our gratitude to our co-guide Ms. Priyadarshini Patil for her active guidance and supervision throughout the process of completion of the project.

We also want to express our sincere thanks to University of Agricultural Sciences, Dharwad for helping us throughout the data collection process.

We would like to thank the entire staff of Data Analytics and Artificial Intelligence Lab for constant support and guidance in the project completion process.

The papers mentioned in the literature survey played an important role in understanding the project requirements in a better way.

We would like to thank our esteemed KLE Technological University for providing an opportunity to take up such a project and gain a great amount of knowledge. We also extend our sincere gratitude to all those who were a part of the project.

Contents

1 INTRODUCTION

1.1 Preamble

In India, maize holds fourth position in the production of grains. Out of the 61 diseases recorded in maize so far, 15 are considered to be contributing largely towards the limited production in maize.

Plant diseases are a threat to food security on a global scale and mainly affect small farmers whose livelihood depends on healthy crops. Manual disease management is a time consuming process. Leaf blight, Common corn rust and Gray leaf spot are the few major diseases in maize. Fall Army worm is one the major pests that infects maize crop.

The area of Artificial Intelligence research has seen an rapid growth in the past few years with respect to applications of Machine Learning. Deep Learning is a category of machine learning that deals with algorithms that mimics the function of the human brain. Among the Deep Learning methods, Convolutional Neural Networks (CNN) has proved to be the best approach to image recognition tasks. CNN learns the features from the training dataset whereas the traditional approaches depend upon the features calculated based on the prior knowledge of the problem.

The dataset developed for this project contains healthy maize leaf images, images of three diseases- Leaf blight, Common corn rust and Gray leaf spot and Ball Armyworm infestation. The aim of the proposed project is to automate the disease classification process in maize and build an application using a deep learning model to classify a given image of a maize leaf among the above five classes.

1.2 Motivation

Maize is one of the important agricultural crops. It is widely used as a source of food for human and animals. In 2014, global production of maize was about 1.17 billion metric tons cultivated on 178 million hectares, wherein the contribution of India was around 24.3 million metric tons. The maize crop is affected by different diseases such as leaf blight, common rust, gray leaf spot, etc. These diseases cause serious loss to the production of maize and causes major loss to the farmers in turn affecting the economy of the country . Hence the early diagnosis of the disease becomes necessary to prevent the financial loss to farmers.

1.3 Objectives of the project

The objective of the project is to develop an effective system using deep learning techniques with an accuracy above 80

1.4 Literature survey

1.4.1 Identification and Recognition of Rice Diseases and Pests Using Convolutional Neural Networks

In this research paper, two state-of-the-art CNN architectures: VGG16 and InceptionV3 have been trained and tested on the collected dataset. The dataset has infected and diseased images that are categorized into 8 different classes. The performance of the architectures was assessed based on fine tuning, transfer learning and training from scratch. Fine tuning the model while training has shown the best results for both the architectures used.

So, a memory efficient CNN model with reasonably good classification accuracy is required. To address the issue large model size, a new training method called two stage training has been proposed. A CNN architecture called a Simple CNN has been proposed in the paper. This model obtained high accuracy compared to two stage training. The analysis made on the architectures shows that the Simple CNN model is better than the architectures such as MobileNet, NasNet Mobile and SqueezeNet on recognizing rice plant diseases and pests.

1.4.2 Identification of Maize Leaf Diseases Using Improved Deep Convolutional Neural Networks

In this paper, two improved deep convolution neural network models are used to classify 9 kinds of diseases in maize crop. The two CNN models used are GoogLeNet and Cifar10. The models used are modified by adjusting the parameters, pool layers, dropout layers and by using RELU optimizer. The obtained results are then compared with those of the original model. The original and improved architecture of GoogLeNet and Cifar10 were used to perform the classification of maize leaf diseases and the analysis is based on accuracy, loss, and other parameters. The proposed method in the paper is compared with the traditional method.

1.4.3 1.4.3 Deep Learning for Classification and Severity Estimation of Coffee Leaf Biotic Stress

This paper describes few of the most common and efficient CNN architectures used for plant disease classification. Some characteristics of the used architectures are listed below.

Name	Parameters(in millions)	Layers
AlexNet	61.8	8
GoogLeNet	6.9	22
VGG19	138	19
ResNet50	25	50

The above mentioned architectures were modified by the addition of a new fully connected layer existing one.

The proposed methodology in the paper consists of a multi-task system based on CNN. The dataset collected for the classification has two tasks, biotic stress classification and severity estimation. The dataset consists of a total of 1747 images of coffee leaves. The dataset includes healthy leaves and diseased leaves that are affected by one or more types of biotic stresses. Two datasets were generated using the obtained images, one with original images and the other containing only symptom images. The training was performed by using pre-trained architectures with the ImageNet dataset. The trained network ResNet50 gave the best results.

1.5 Problem definition

To propose a classification system using deep learning techniques that classifies the diseases and pest infestation in maize crop.

2 PROPOSED SYSTEM

2.1 Description of Proposed System

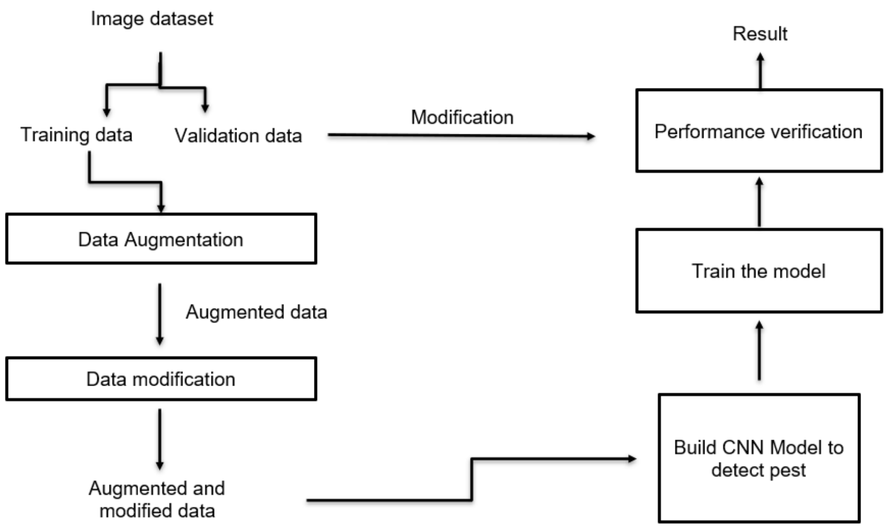


Figure 1: Proposed System

The figure 1. shows how the proposed system is designed. An android application is designed for live detection using VGG19 architecture. The application starts accepting the live image as soon as the application is opened. The identified image must belong to one of the classes - Leaf Blight, Common Corn Rust, Gray Leaf Spot, Fall Armyworm infestation or Healthy. The percentage value for each class is displayed and the class holding the highest percentage value is the class to which the image belongs to. The application is supported by the android versions 4.0 and above.

2.2 Description of Target Users

The application can be used by Farmers and people associated with agriculture sector.

2.3 Advantages/Applications of Proposed System

The application detects the type of disease or pest infestation in maize crop. The application is user friendly and saves time that is required to manually detect the disease or pest infestation. There are exists methods for pest detection such as remote sensing which involves placing sensors in the field but the designed application helps to avoid the expenses as no extra devices or sensors are required.

2.4 Scope

The application can be used in the field of agriculture for other crops also if provided with the appropriate data. As per further application, user can get to know the type of disease or infestation caused way before the extreme damage is caused to the crop. This way, frequent damage being caused to the crops cultivated on hectares of area can be reduced as the disease is detected well in advance.

3 SYSTEM SOFTWARE SPECIFICATION

3.1 Overview of SRS

SRS describes the nature of the project. This is important to know the overall behavior of the system. It briefly mentions the functional and non-functional requirements of the project and also states the software and hardware specifications. It also describes the use case descriptions of our project and helps us understand the scope of our project.

3.2 Requirement Specification

3.2.1 Functional Requirements

Functional requirements of a system are the system features and they focus on user requirements. Functional requirements describe the services that the system must offer. They help you capture the desired behavior of the system. They are basically what the system must do or must not do. The functional requirements of the proposed system are:

1.The system shall be able to accept/read an image fed to the system.
2.The system shall be able to generate the output on given image i.e.identify pest infestation or diseased.
3.The system shall be able to distinguish between healthy plant and the diseased ones.

3.2.2 Use case diagram

Figure 2 represents the use case diagram of the system designed. The three main actors are farmer, researcher and the system(model). The application can be used by both researcher and farmer. The user shall be able to open the application on their android device which takes live preview of objects. The input image has to be clear enough for the model to detect the disease or the pest infestation. It must also be able to detect if the plant is healthy.

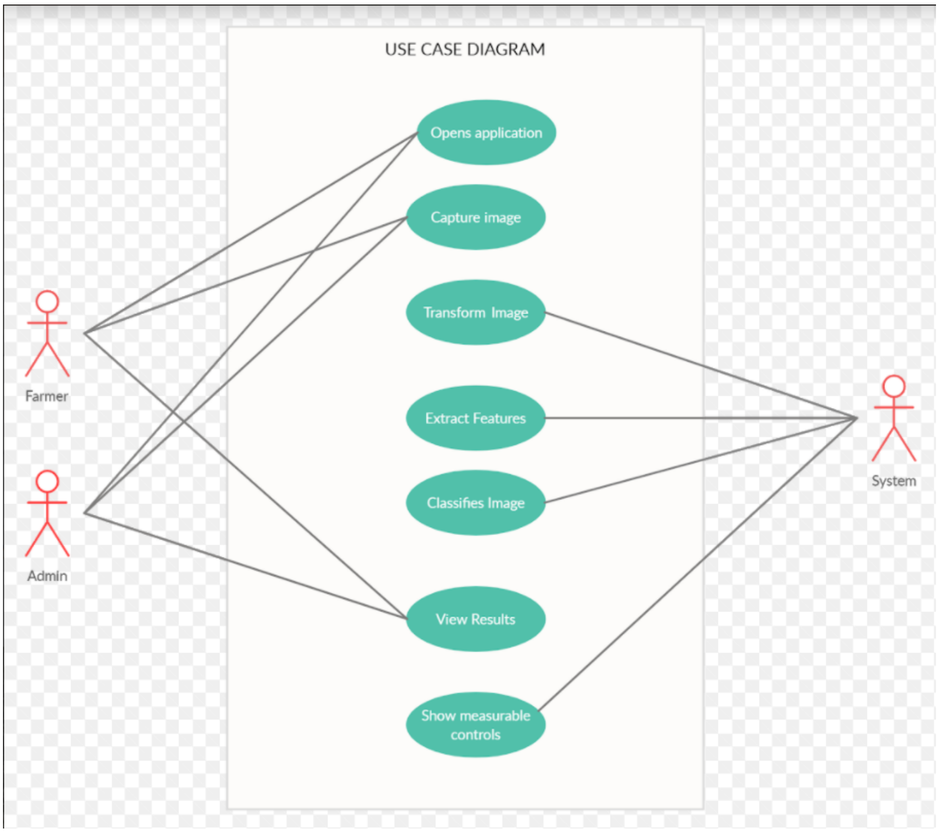


Figure 2: Use Case Diagram

USE CASE 1: Check Plant Disease
ACTORS: Farmer
PRE-CONDITION: The system is built and is already in use.
POST-CONDITION: Identifying the type of pest infestation.
MAIN SUCCESS SCENARIO: **1.** Farmer will click a picture. **2.** The model will reads it as input. **3.** The system should show the output.
EXCEPTION SCENARIO: **1.** System Failure: System unable to start. **2.** Giving a low quality image as input due to low camera quality. **3.** Error in output.

Figure 3: Use Case Description

3.2.3 Use case descriptions

3.2.4 Non Functional Requirements

1.The response time of the system should be less than 10s.
2.The system should provide an accuracy more than 80 percentage.

3.3 Software and Hardware requirement specifications

The system will only work on devices with Android 4.0 and above.

4 SYSTEM DESIGN

4.1 Architecture of the system

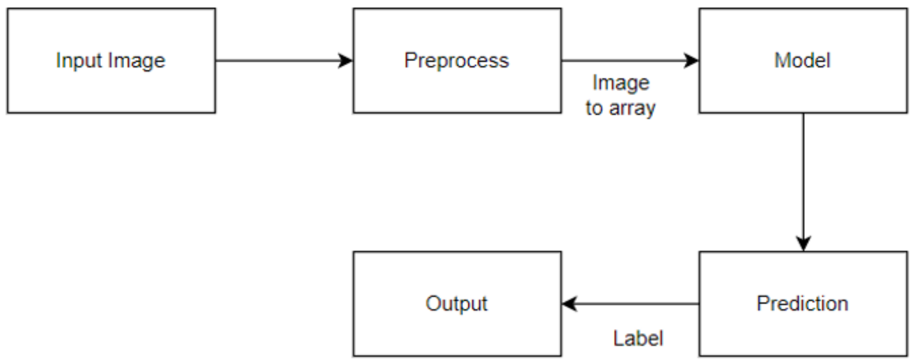


Figure 4: Architecture Diagram

Figure 3 represents the architecture of the proposed system. An image of leaf of a maize crop is given to the system. The image undergoes preprocessing before it is given as an input to the built deep learning model ,that is, the image is reshaped and rescaled. The model predicts whether the given image is an image of leaf of maize crop that is infected with Ball Armyworm, an image of a diseased maize crop or a healthy leaf.

4.2 Low Level Diagram

Low level diagram of the system includes a brief description of the working of the model.It consists of the various modules included in the system. The low level diagram is the basic diagram enough to explain the outine of the model.As seen in fig 4, we see that the application takes live preview of the input image, preprocesses the image and sends it to the model which outputs the probablity of the input image in various classes .

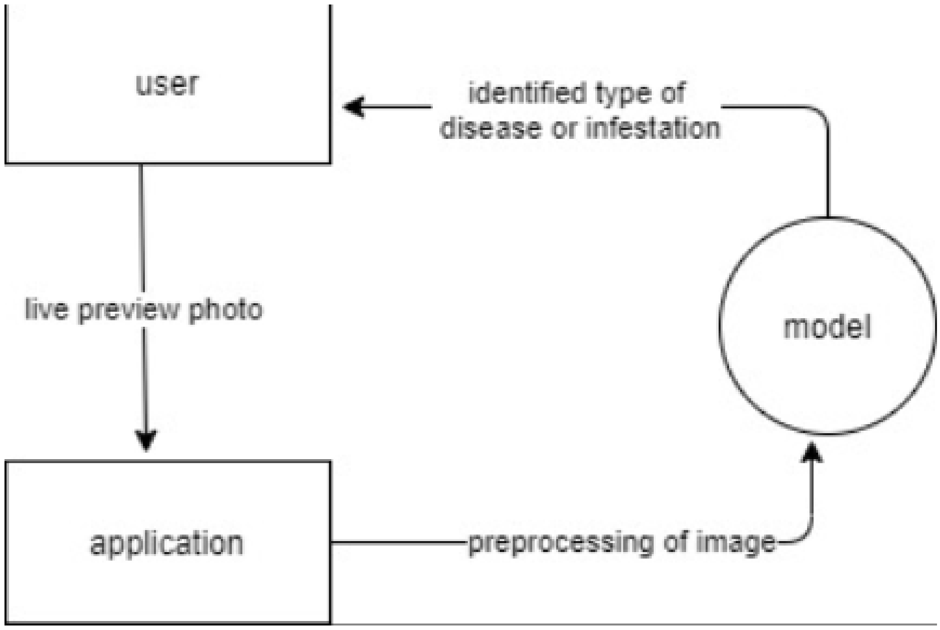


Figure 5: Low Level Diagram

4.3 Detailed Diagram for the proposed system

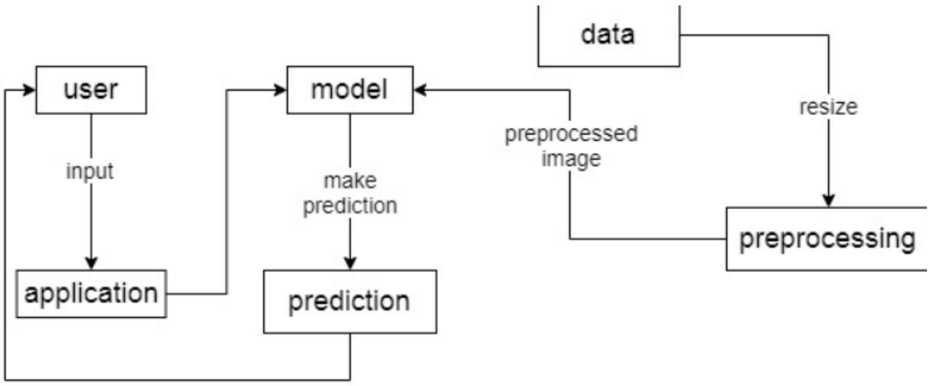


Figure 6: Detailed Diagram for Propsed System

High level Diagram explains the working of the system in detail .In fig 3, we can see a more abstract working of the system. It explains how the user live previews the input image and sends it to the application. The models gives the probability of the input image in various classes. When the application live previews the image, it first rescales it to 224 x 224 which is the input image size in the model. This is the pre-processing step mentioned.

4.4 Activity Diagram

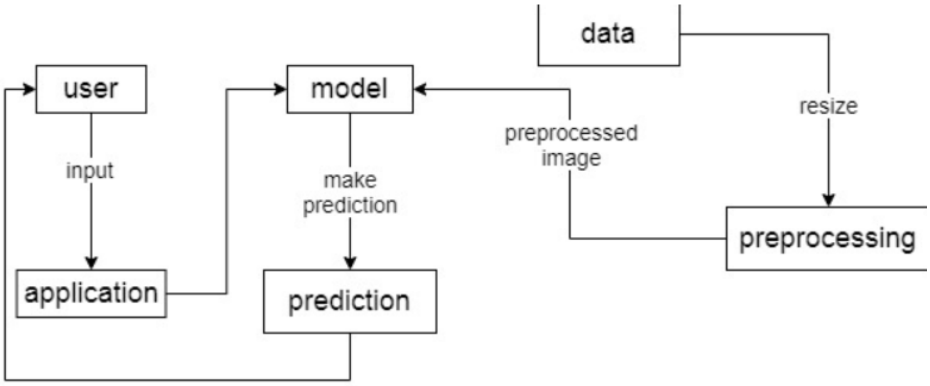


Figure 7: Activity Diagram

An activity diagram visually presents a series of actions or flow of control in a system similar to a flowchart or a data flow diagram.
The following are the steps given:
1. The user live preview the input image
2. The photo has to be clear in order to be processed. So the user has to wait till the photo is clear.
3. The photo is processed as soon as it is clear.
4. The photo is then the input to the model where its probability in various class is found
5. If the photo does not belong to any class, it is classified as ”others”.
6. The application outputs the probability of the input images in various classes.

4.5 Dataset Description

The dataset consists of a total of 2,546 images of maize leaves. The dataset has diseased maize leaf images, images with pest infestation by Ball Armyworm and images of healthy maize leaves. The diseases include Leaf Blight, Common Corn Rust and Gray Leaf Spot. The pest infected images were collected from the field while the diseased images were gathered from websites like Kaggle.

The dataset is further divided into train and test set. The train set consists of 80 percentage of the total dataset while test set contains 20 percentage of the total dataset. The train dataset is further divided into train and validation set.

5 IMPLEMENTATION

5.1 Proposed Methodology

Image Dataset :
Images of leaves of maize plant are given to us which are affected by diseases such as corn blight, grey leaf spot and common rust. The dataset also includes images which are affected by pests and healthy leaf images. Due to less number of images in dataset we applied data augmentation in order to increase images for training as well to avoid under and over-fitting. We are using Keras python library for doing so and the augmented images are resized, zoomed, rotated and filled with reverse images in voids with a batch size of 32.

The 20 percentage of train set is set aside for validation set. The sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyperparameters.

Data Augmentation:
More the data, the better the performance of the model. The collected dataset has a total of 2,546 images. To increase the number of images in the dataset, data augmentaion is applied. This technique also prevents the model from over-fitting. We use Keras python library for doing so. Data augmentation includes resizing, zooming, rotating images etc and creating new images.

CNN Architecture:
A Convolutional Neural Network is a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various objects in the image and be able to differentiate one from the other. The structure of CNNs consisted of three main substructures, which include: convolutional layers, pooling layers, fully connected layers.

Figure 8. represents the structure of CNN. Convolutional layers are made from several feature maps, and each unit of feature maps is made from convolving a small region in input data which is called the filter/kernel. A new feature map is created by sliding a kernel over the input. The convolution can be used in various kinds of data such as image, text. Unlike the standard neural network, each neuron in the layers is not connected to all of the nodes (neurons) in the previous layer but is connected to just

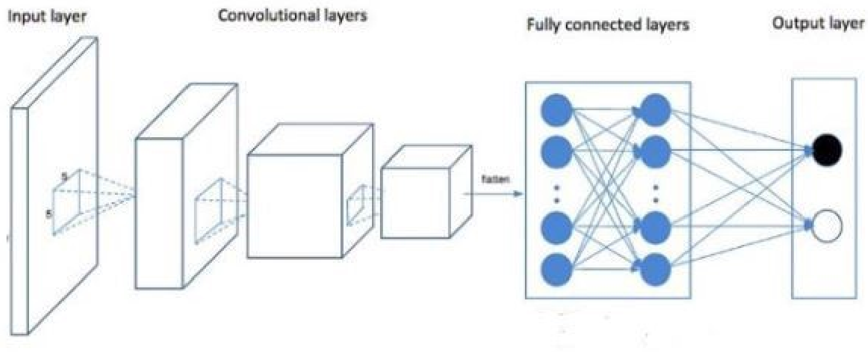


Figure 8: Structure of CNN

small number of nodes.

Pooling layers are commonly used immediately after convolutional layers. These layers were generated to simplify the information and reduce the scale of feature maps. In other words, pooling layers make a condensed feature map from each feature map in convolutional layers. Max-pooling and average-pooling are two of the most prevalent processes. The pooling layers are necessary to reduce the computational time and overfitting issues in the CNN.

Fully connected layers are the final layers in the CNN structure that can be one or more layers and placed after a sequence of convolution and pooling layers. This part of CNN comprises the composite and aggregates of the most important information from all procedures of CNN. Consequently, these layers provide the feature vector for the input data, which can be used for some machine learning tasks such as classification, prediction. The last layer of fully connected layers is known as softmax classifier and determines the probability of each class label over N classes.

Pre-trained CNN models are trained by a huge number of the image with the aim of detection and classification data (images) in a large number of classes. AlexNet, GoogleNet, SqueezNet, ResNet, DenseNet-201, Inception-v3, and VGG16/19 are some of the more prevalent pre-trained models used in transfer learning technique.

We have used VGG19 as our CNN learning model which provided us with faster learning and better accuracy than the other models we implemented. Pre-trained weights from VGG19 are used via transfer learning to train the model. On average, using cross-validation, the proposed VGG19 model achieved the best accuracy rate of 87.78 percentage for all classes. The test accuracy for our final model was 79.40 percentage The other implemented models were resnet50, InceptionV3, and a 5 layered CNN with test accuracies 68.40, 70.40 and 78.23 respectively. In VGG19 an additional layer was added to regularize the model to avoid over-fitting. The model was then converted into a tflite file in order to import and embed the model into an android app. The app takes the live feed using the camera and predicts a specific class to which the input image may belong to. On testing over more than 100 samples we got more than 76 percentage satisfactory results.
Architecture of VGG19:
VGG19 is a model which consists of 19 layers (16 convolution layers, 3 Fully connected layer, 5 MaxPool layers and 1 SoftMax layer). There are other variants of VGG like VGG11, VGG16 and others. AlexNet came out in 2012 and it improved on the traditional Convolutional neural networks, So we can understand VGG as a successor of the AlexNet but it was created by a different group named as Visual Geometry Group at Oxford's and hence the name VGG.

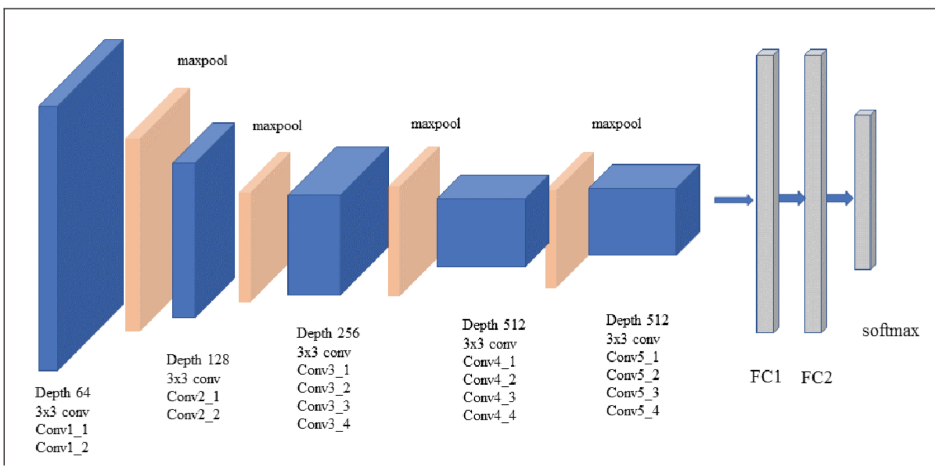


Figure 9: Architecture of VGG19

Figure 9. represents the layers in VGG19. It has 16 convolution layers, 3 fully connected layers, 5 max pool layers, and a softmax layer.

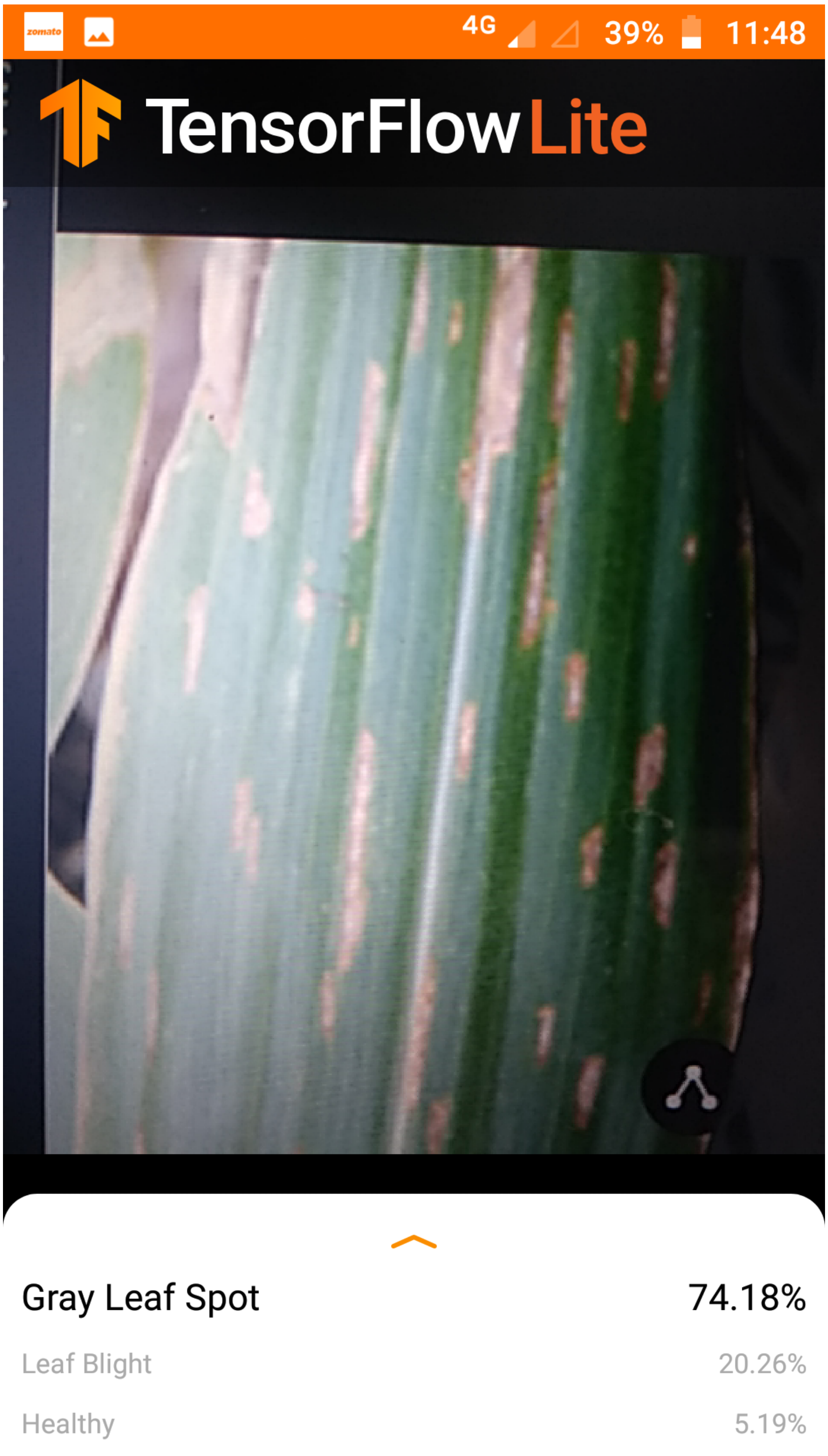
A fixed size of (224 * 224) RGB image was given as input to this model. The input images are rescaled and value of every pixel ranges from floating point value 0 and 1. Kernels of (3 * 3) size with a stride size of 1 were used in the convolution layers. Padding was used to preserve the spatial resolution of the image. Max pooling was performed over a 2 * 2 pixel windows with stride 2. This was followed by Rectified linear unit(ReLu) to introduce non-linearity to make the model classify better and to improve computational time. We implemented three fully connected layers from which first two were of size 4096 and after that a layer with 6 channels for 6 class classification. The final layer has a softmax function. L2 regularization was applied with a learning rate of 0.0004 in order to avoid overfitting.

The model was trained and testes on the dataset. It obtained an accuracy of 80 percentage on the test set.

5.2 Description of Modules

1. Gray Leaf Spot
Input:

Output:


4G 39% 11:48
TensorFlow Lite
Gray Leaf Spot 74.18%
Leaf Blight 20.26%
Healthy 5.19%

2. Common Rust

Input:

Output:

3. Infected
Input:

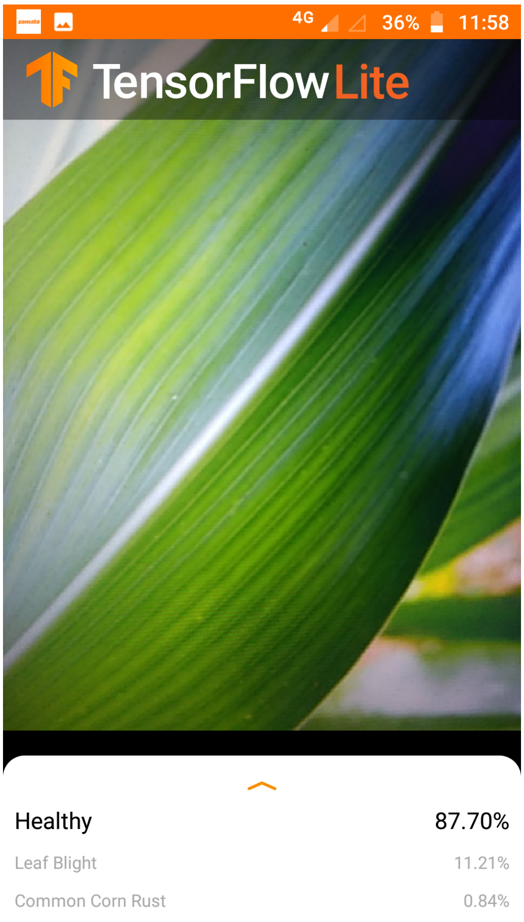

©MARLIN E. RICE

Output:


4G 38% 11:54
TensorFlow Lite
Infected 95.23%
Healthy 2.79%
Common Corn Rust 1.37%

4. Healthy
Input:

Output:



5. Blight
Input:

Output:


4G
44%
11:37
TensorFlow Lite
Leaf Blight
50.43%
Infected
44.25%
Gray Leaf Spot
5.30%

6 TESTING

6.1 Acceptance Testing

Test Id	Input	Expected Output	Actual Output
1	Healthy leaf image	Healthy	Healthy
2	Pest Infected Image	Infected	Infected
3	Leaf with blight disease	Leaf Blight	Leaf Blight
4	Leaf with common rust disease	Common Corn Rust	Common Corn Rust
5	Leaf with gray leaf spot disease	Gray Leaf Spot	Gray Leaf Spot

7 RESULTS DISCUSSIONS

Initially the model used for training the available dataset was RESNET50 with an accuracy of 68.40 percentage. To improve the accuracy of the system, other models implemented were Inception V3 and VGG19. We also implemented our own model but the accuracy observed was 70.40 percentage for Inception V3, 78.23 percentage for our model and 79.40 percentage for VGG19.

8 CONCLUSION AND FUTURE SCOPE

A target recognition method based on improved VGG19 for quickly and accurately detecting diseases or infestation in live image was proposed. A pre-trained VGG19 architecture has been implemented using Keras framework and achieved successful feature extraction. The experimental results based on test accuracy and validation loss on current available data shows that the model is faster and accurate.

However, still some issues can be identified in this proposed method. This method performance can be further improved by expanding the database which can be done by collecting images regarding various diseases in the maize, manually in the future.

9 References

[1] Identification and Recognition of Rice Diseases and Pests Using Convolutional Neural Networks:
https://arxiv.org/pdf/1812.01043.pdfhttps://arxiv.org/pdf/1812.01043.pdf

[2] Deep Learning for Classification and Severity Estimation of Coffee Leaf Biotic.:
https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8374024tag=1

[3] Deep Learning for Classification and Severity Estimation of Coffee Leaf Biotic:
https://arxiv.org/pdf/1907.11561.pdf