PLANT DISEASE DETECTION

A Project work submitted in partial fulfillment of the requirement for the award of the degree of

BACHELOR OF TECHNOLOGY

in

ELECTRONICS & COMMUNICATION ENGINEERING

by

M. Nikhitha Reddy
Regd.No: 17211A04E3

M. Sowmya
Regd.No: 17211A04C1

M. Saipriya Reddy
Regd.No: 17211A04E7

M. Gopala Krishna Chowdary
Regd.No: 17211A04C7

Under the esteemed guidance of

Mr. Jigar Patel
Assistant Professor

B.V. Raju Institute of Technology

UGC- AUTONOMOUS
Department of Electronics and Communication Engineering
Vishnupur, Narsapur, Medak.(Dt)
(Affiliated to JNTU, Hyderabad)
2021

B.V.Raju Institute of Technology
Vishnupur, Narsapur, Medak.(Dt) Pin:502313
(Affiliated to JNTU, Hyderabad)
Ph: 08458-222000, 222001 Fax: 08458-222002

Department of Electronics & Communication Engineering

CERTIFICATE

This is to certify that the Project work entitled PLANT DISEASE DETECTION is being submitted by Ms.M.Nikhitha Reddy, Ms.M.Sowmya, Ms.M.Saipriya Reddy Mr.M.Gopala Krishna Chowdary in partial fulfillment of the requirement for the award of the degree of **B.Tech. in Electronics & Communication Engineering**, by Jawaharlal Nehru Technological University Hyderabad is a record of bonafide work carried out by him under my guidance and supervision from 2020 to 2021

The results presented in this project have been verified and are found to be satisfactory.

Mr. Jigar Patel
Assistant Professor

Dr.Sanjay Dubey
Professor & HOD, Dept. of ECE.

EXTERNAL EXAMINER

CENTER FOR EXCELLANCE

IN

ARTIFICIAL INTELLIGENCE AND MACHINE LARNING

CERTIFICATE

This is to certify that **M.Nikhitha Reddy, M.Sowmya, M.SaipriyaReddy, M.Gopala Krishna Chowdary** bearing Roll No: **17211A04E3, 17211A04C1, 17211A04E7,17211A04C7.** respectively has successfully completed training on AI & Machine Learning and implemented a project tilted "**PLANT DISEASE DETECTION"** in Centre for Excellence In Artificial Intelligence and Machine Learning, Department of ECE, B.V.Raju Institute of Technology, Narsapur, during the period from 2020 to 2021.

Coordinator	HOD
Dr. Syed Abudhagir U., Ph.D	Dr.Sanjay Dubey.,Ph.D

ACKNOWLEDGEMENTS

We take this opportunity to express my indebted gratitude to the persons who contributed for our work, for being an inspiration and guide which led to the successful completion of the project.

We are grateful towards the college management and the beloved principal Dr. K. Lakshmi Prasad for providing the necessary infrastructure and facilities that ensure smooth and satisfactory execution of the project. We would like to express profound gratitude to the head of the department Dr.Sanjay Dubey.,Professor & HOD,Dept. of ECE, for his encouragement, inspiration and close monitoring and guidance he gave during the execution of the project.

We express our sincere thanks to our guide Mr. Jigar Patel, Assistant Professor, Dept. of ECE, for his valuable suggestions and motivation in successful completion of project. We also wish to express our thanks to all the faculty members and laboratory staff who were helpful directly and indirectly for the completion of the project.

Name	
M. Nikhitha Reddy	17211A04E3
M. Sowmya	17211A04C1
M. Saipriya Reddy	17211A04E7
M. Gopala Krishna Chowdary	17211A04C7

DECLARATION

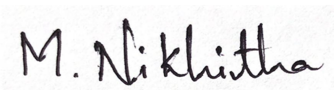
We hereby declared that the project work entitled "PLANT DISEASE DETECTION " submitted to the JNTU Hyderabad and Department of electronics and communication engineering, BVRIT Narsapur, is a record of an original work done by that **M.Nikhitha Reddy, M.Sowmya, M.SaipriyaReddy, M.Gopala Krishna Chowdary** bearing Roll No: **17211A04E3, 17211A04C1, 17211A04E7,17211A04C7** during the period 2020 to 2021 under the guidance of **Mr. Jigar patel**, Assistant Professor , Department of ECE and this project work is submitted in the partial fulfillment of the requirements for the award of the degree "Bachelor of Technology" in "Electronics & Communication Engineering".

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree.

1)Name of the Student

M. Nikhitha Reddy

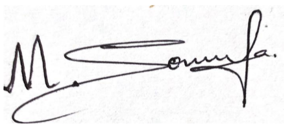
Signature



2) Name of the Student

M. Sowmya

Signature



3) Name of the Student

M.Gopala Krishna Chowdary

Signature

4) Name of the Student

M. Saipriya Reddy

Signature

ABSTRACT

Plant disease is an ongoing challenge for smallholder farmers, which threatens income and food security. Rapid human population growth requires corresponding increase in food production. Easily spreadable diseases can have a strong negative impact on plant yields and even destroy whole crops. That is why early disease diagnosis and prevention are of very high importance. The recent revolution in smartphone penetration and computer vision models has created an opportunity for image classification in agriculture. Convolutional Neural Networks (CNNs) are considered state-of-the-art in image recognition and offer the ability to provide a prompt and definite diagnosis. In this paper, the performance of a pre-trained model in detecting crop disease is investigated. The developed model is deployed as a web application and is capable of recognizing plant diseases out of healthy leaf tissue. A dataset containing 20,638 leaf images; captured in a controlled environment, is established for training and validating the model. Validation results show that the proposed method can achieve an accuracy of 98.3%. This demonstrates the technical feasibility of CNNs in classifying plant diseases and presents a path towards AI solutions for small holder farmers. The user, who is the Farmer clicks a picture of the crop and uploads it to the server by using the android application installed in mobile or by using webpage. The uploaded image is processed and accordingly the features of that image are extracted. Based on those features the classification of image is done using Convolutional neural network and the classes having maximum probability is selected. Then the result consisting of the disease name is retrieved. This result is then uploaded into the message table in the server and retrieved in mobile application or on the webpage where corresponding information such as pesticide name, amount of pesticide to be used and organic pesticides which are stored. Now the Farmer will be able to retrieve the complete information in a presentable format on the screen of the application.

CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In India, Agriculture is the main source of income. Farmers grows varieties of crops based on their requirement. Since the plants suffer from the disease, the production of crop decreases due to infections caused by several types of diseases on its leaf, fruit, and stem. Leaf diseases are mainly caused by bacteria, fungi, virus etc. To overcome this, diseases in leaves are classified based on the diseased leaf types using Neural Networks algorithm and thus can take necessary steps in time to minimize the loss of production. In this Idea, Farmer take the picture of the leaf of crop which he has sown in his Farm. After clicking it will be uploaded on server and then uploaded image is processed and accordingly the features of that image are extracted. Based on those features the classification of image is done using Neural Networks then the result consisting of the disease name is retrieved .

1.1.MOTIVATION:

There are a variety of diseases that affect plants, which can each cause economic, social, and ecological loss. In this context, a timely and accurate diagnosis of plant diseases plays an important role in preventing the loss of productivity and quantity of agricultural products. Detection of plant diseases is usually performed manually. Such processes are conducted by experts such as botanists and agricultural engineers, first by visual inspection and later in a laboratory environment. These traditional methods are often time-consuming and complex processes . For these reasons, it has become important to automatically identify diseases based on image processing and machine learning. Automatic plant disease diagnosis with visual inspection can be of great benefit to users who have little to no knowledge of the product they are cultivating.
There have been numerous studies in the literature with regards to the detection of plant diseases. In the past decade, these studies have often been conducted based on the classification process by using features such as color, shape, and texture.

In plants, disease indications usually occur on leaves, fruit, buds, and young branches. This situation causes fruit to be wasted (to drop) or be damaged. In addition, these diseases lead to the formation of new infections and the spread of the disease for reasons such as seasonal conditions. For this reason, it is very important to determine the disease in advance and to take the necessary precautions before it spreads to other trees. As a result, the fight against diseases and pests in plants is the single most important issue in agriculture.

1.2.OBJECTIVE:

Plant diseases cause a major production and economic losses in the agricultural industry. The disease management is a challenging task. Usually the diseases or its symptoms such as colored spots or streaks are seen on the leaves of a plant. In plants most of the leaf diseases are caused by fungi, bacteria, and viruses. The diseases caused due to these organisms are characterized by different visual symptoms that could be observed in the leaves or stem of a plant. Usually, these symptoms are detected manually. With the help of image processing, Automatic detection of various diseases can be detected with the help of image processing. Image processing plays a crucial role in the detection of plant diseases since it provides best results and reduces the human efforts. The image processing could be used in the field of agriculture for several applications. It includes detection of diseased leaf, stem or fruit, to measure the affected area by disease, to determine the color of the affected area. Cultivation is one of the most remunerative farming enterprises in India. The naked eye observation by the experts is approach usually taken in identification and detection of plants. This approach is time consuming in huge farms or land areas. The use of image processing techniques in detection and identification of plant diseases in the earlier stages and thereby the quality of the product could be increased . These systems monitor the plant such as leaves and stem and any variation observed from its characteristic features, variation will be automatically identified .

1.3.SCOPE:

Following the discovery of the causes of plant diseases in the early twenty first century, growing understanding of the interactions of pathogen and host has enabled us to develop a wide array of measures for the control of specific plant diseases. From this accumulated knowledge base, we can distil some general principles of plant disease control that can help us address the management of new problems on whatever crop in any environment. Automatic detection of plant diseases provides benefits in monitoring large fields of crops, and thus automatically detects the diseases from the symptoms that appear on the plant leaves. This enables machine vision that is to provide image based automatic inspection from the advent Digital Image Processing many people have tried and classified diseases using many techniques. To detect the crop disease we have studied many algorithms, it involved an application of Convolutional Neural Networks (CNN) with a customized architecture, in the leaf disease from there leaves.

CHAPTER 2

LITERATURE SURVEY

Wan Mohd Fadzil et al. [1], mentioned a method that used for detecting disease which occurs on leaves of orchid plant. Images of orchid plant leaflet are obtained utilizing digital camera. For categorizing images into two disease class, aggregate of several strategies like morphological processing, filtering technique , and border segmentation method are used by the algorithm.. two classes used in this are solar scorch and black leaf spot. However, the segmentation technique proposed and used in this can only distinguish two different types of orchid leaf disease. For classification of other types of leaf disease present on orchid, new or other segmentation technique have to develop. This is because there need many combination of the processing techniques to find robust for border segmentation techniques.

Aditya Parikh et al [2] primary focuses on detection of disease and also on estimating disease stage for a given image of cotton plant leaf. The proposed work uses two cascaded classifiers, first classifier segments leaf from the background for which local statistical features are used .Then another classifier is trained using luminance and hue from HSV color space so that classifier can detect disease and identify its level. The algorithm that has been developed is universal, as it can be applied to any disease. However , cascaded classifiers depends on various conditions i.e. border of the leaves are viewable, Leaves are big size for analysis and the probing requires controlled environment.

Bhumika S.Prajapati et al [3] presents a survey on cotton leaf disease detection and classification. It is difficult for human eyes to identify exactly which type of leaf disease exists on the plant leaf. Therefore, the usage o f machine learning technique and image processing technique can be helpful to accurately identify the cotton leaf disease s. The images which are used for this task were acquired using digital camera from the cotton field . In order to remove background from the image the background removal technique is applied in pre-processing step. Then, the background removed images are processed further for image segmentation which is done by otsu thresholding technique. However, this work describe only general and different approach which detect and classify leaf diseases of cotton and describe segmentation as well as background removal techniques.

P. R. Rothe et al [4] presents a pattern recognition system which identify also classify cotton leaf diseases which are Bacterial_Blight, Alternaria and Myrothecium. The pictures taken for this purpose are taken from the cotton fields in Buldana and Wardha district and the fields at "Central Institute of Cotton Research Nagpur". For image segmentation active contour model is used and for the training of adaptive neuro-fuzzy inference system , Hu's moments are extracted as features. However, seven invariant features are extracted from 3 types of diseased_ leaves images and this is done to train neuro-fuzzy inference neural network. Neural network classification is depends on invariant features.

Melike Sardogan et al [5] presents a CNN algorithm and Learning Vector Quantization algorithm based method for leaf disease detection and categorization of tomato plant. The dataset contains 500 pictures of tomato plant leaves with four symptoms of diseases. They have modeled a CNN so that automatic feature extraction and classification is done. However, for this study one of the main challenges is that the leaves having different diseases are fundamentally the same as one another. Accordingly, this likeness can make a few leaves be collapsed into to wrong classes.

Norfarahin Mohd Yusoff et al [6] gives a real-time technique of detection of edge for identifying diseases present on Hevea leaves (rubber leaves) and also its hardware implementation. There are main three diseases which occurs on Hevea leaves. for image comparison these three diseases which are Bird's Eye Leaf Spot ,Corynespora Leaf Spot and Collectotrichum Leaf Disease are used. The disease can be detected by using Sobel edge detection algorithm . The result generated by "FPGA Cyclone IV E ".this result is then shown using a monitor. Sobel edge detection algorithm is produce with MATLAB. Result of both techniques are compared. However, Sobel edge_detection algorithm execution depends on MATLAB and FPGA hardware and display the output on the VGA monitor.

Indumathi.R et al [7] discovers the affected area of the leaf, and the disease that attack the leaf as well. This is done with the use of Image Processing. this system make use of "K Medoid clustering" and the "Random Forest algorithm" to produce greater precision in the leaf disease detection. pre-processing is performed on image, and then the method of clustering is applied to identify the region of the leaf affected. However, Random Forest algorithm is decision tree based algorithm. Accuracy is low compared to other algorithms. Basically random forest used with text data.

Gayatri Kuricheti et al [8] produce an algorithm for detection and prevention of spreading of diseases to the entire crop and results in excellent harvest creation.. The database of various leaf images was created. These images are processed using k-Means image segmentation and textural analysis of leaf images was completed utilizing GLCM. SVM classifier is utilized to categorize the feature extracted images after ranking their attributes using an information gain algorithm. However, the fundamental drawback of K-mean clustering algorithm is that the number of clusters are required to fix . For the extracting infected part of the leaf , 3 clusters can be ideal .Each cluster will denote background in image, healthy part of leaf and the infected part of leaf separately.

Chaowalit Khitthuk et al [9] present a diagnosis system for plantleaf disease utilizing unsupervised neural network. Color features as well as texture features are used for processing images. The system is primarily made of two processes: extraction of features and classification of disease. The method of extraction of the disease feature analyzes the presence of features using statistic-based gray level co-occurrence matrix and texture feature equations. The method of classifying diseases deploys the "unsupervised simplified fuzzy ARTMAP neural network" to categorize disease types. Four forms of photographs of grape leaf disease are used to check the classification efficiency of the method which is rust, downy_mildew, scab and no disease. However in many classifications systems unsupervised feature isn't practically suitable comparing to traditional backpropagation network and machine learning.

PENG JIANG et al [10] presented the collection of images of apple leaves which are affected by disease under real field conditions. Complex and laboratory images are used. The dataset is first created via image annotation and data_augmentation technologies. Based on this, a new model that uses deep-CNNs for apple leaf disease detection is proposed by introducing Rainbow concatenation and the Google Net Inception structure. Finally, under the hold-out testing dataset, using a dataset having 26,377 images of diseased apple leaves, the proposed model is trained. This model can be used detecting common apple leaf diseases such as Grey spot, Brown spot, Alternaria leaf spot, Rust and Mosaic . However in this failures of disease detection are observed.

CHAPTER 3

ANALYSIS AND DESIGN

A Convolutional Neural Network (CNN) is specific type of neural network. In this paper, we choose to simplify its presentation by considering that it can be decomposed into two parts: a feature extraction part and a classification part (see Figure 1-a). Features selection aims at extracting information from the input to help the decision-making. To select features, a CNN is composed of n3 stacked convolutional blocks that correspond to n2 convolutional layers (denoted γ), an activation function (σ) and one pooling (denoted δ) layer [ON15]. This feature recognition part is plugged into the classification part of n1 Fully-Connected (FC) layers (denoted λ). Finally, we denote s the softmax layer (or prediction layer) composed of |Z| classes. To sum up, a common convolutional network can be characterized by the following formula:

$$s \circ [\lambda]\ n1 \circ [\delta \circ [\sigma \circ \gamma]\ n2\ ]\ n3\ .$$

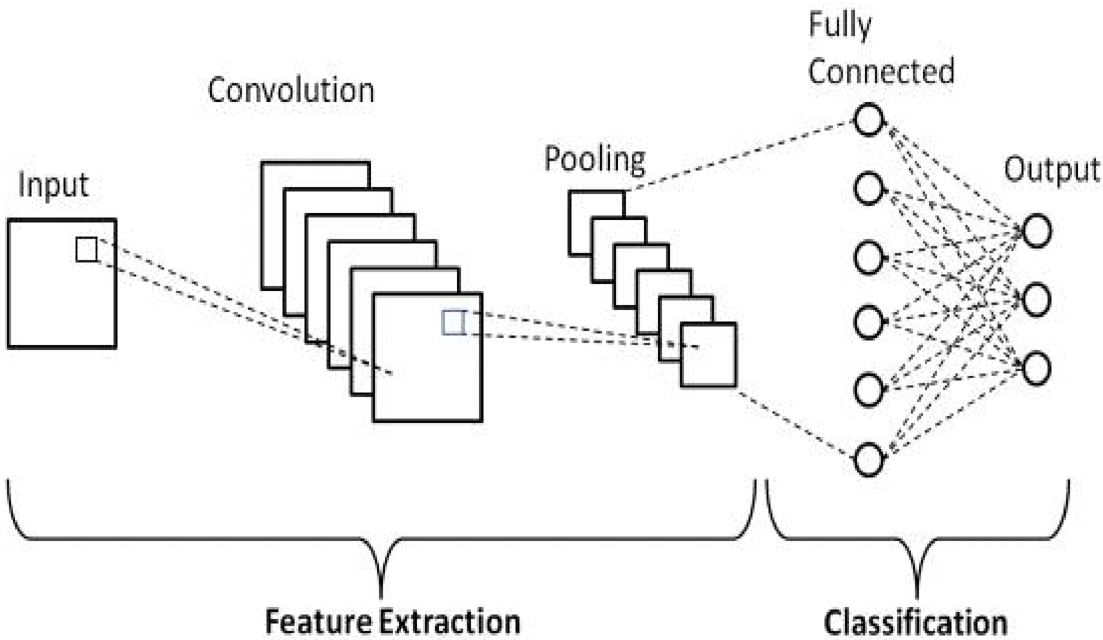


Figure 3.1: Basic Architecture of CNN

Convolutional layer:

The convolutional layer is the core building block of a CNN, and it is where the majority of computation occurs. It requires a few components, which are input data, a filter, and a feature map. Let’s assume that the input will be a color image, which is made up of a matrix of pixels in 3D. This means that the input will have three dimensions—a height, width, and depth—which correspond to RGB in an image. We also have a feature detector, also known as a kernel or a filter, which will move across the receptive fields of the image, checking if the feature is present. This process is known as a convolution.

The feature detector is a two-dimensional (2-D) array of weights, which represents part of the image. While they can vary in size, the filter size is typically a 3x3 matrix; this also determines the size of the receptive field. The filter is then applied to an area of the image, and a dot product is calculated between the input pixels and the filter. This dot product is then fed into an output array. Afterwards, the filter shifts by a stride, repeating the process until the kernel has swept across the entire image. The final output from the series of dot products from the input and the filter is known as a feature map, activation map, or a convolved feature.

Pooling layer:

The pooling layer is a non-linear layer that divides the dimension of the input such that the most relevant information is preserved. To apply its down sampling function, a pooling window and a pooling stride have to be configured. Usually, these variables have the same value to avoid overlapping. The window slides through the input values to select a segment to be applied to the pooling function. In deep learning, two pooling functions are commonly used:

• MaxPooling – The output is defined as the maximum value contained in the pooling window.

• AveragePooling – The output is defined as the average of the values contained in the pooling window. Figure 1-b shows an example of this function applied to a 1-D input with pooling_window = pooling_stride = 2.

Fully-Connected Layer

The name of the full-connected layer aptly describes itself. As mentioned earlier, the pixel values of the input image are not directly connected to the output layer in partially connected layers. However, in the fully-connected layer, each node in the output layer connects directly to a node in the previous layer.

This layer performs the task of classification based on the features extracted through the previous layers and their different filters. While convolutional and pooling layers tend to use ReLu functions, FC layers usually leverage a softmax activation function to classify inputs appropriately, producing a probability from 0 to 1.

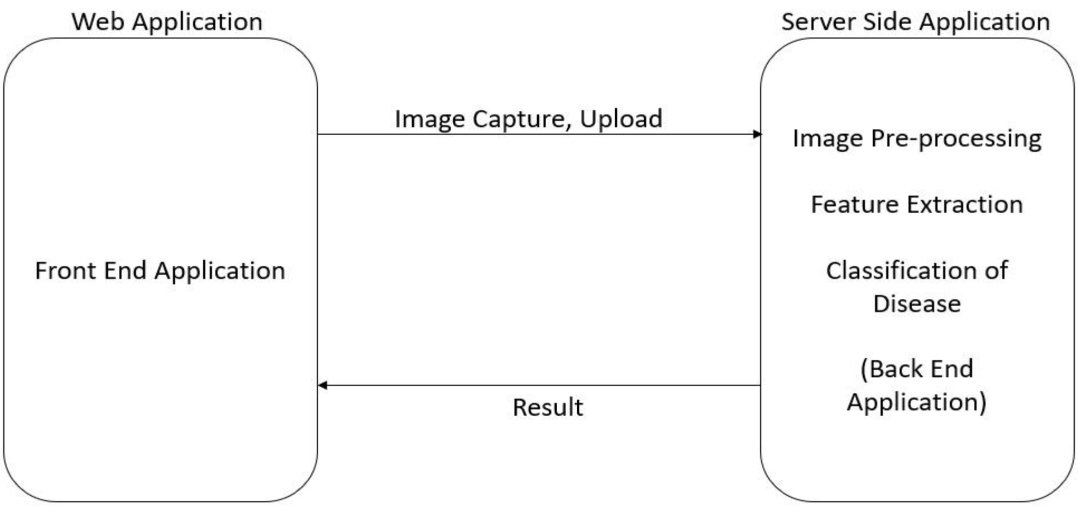


Figure 3.2: Structure of Proposed System

Summary:

The process of diagnosis of leaf diseases involves many tasks, such as image_acquisition, preprocessing of image , extraction of features from image and classifying leaf diseases is depends on image feature that is color features, shape features and texture features. The first stage is the image acquisition . In this phase, image is uploaded from the images of the leaf dataset. Then the preprocessing on image is performed using different techniques. In the third phase, extraction of features is done from the picture for the part of the leaf which is infected.This is done on the basis of particular properties be tween pixels in the image or their texture Then to classify the features which are represent the given image statistical analysis tasks are performed. machine learning is used to compare image features. Finally, classification result

Chapter 4

IMPLEMENTATION

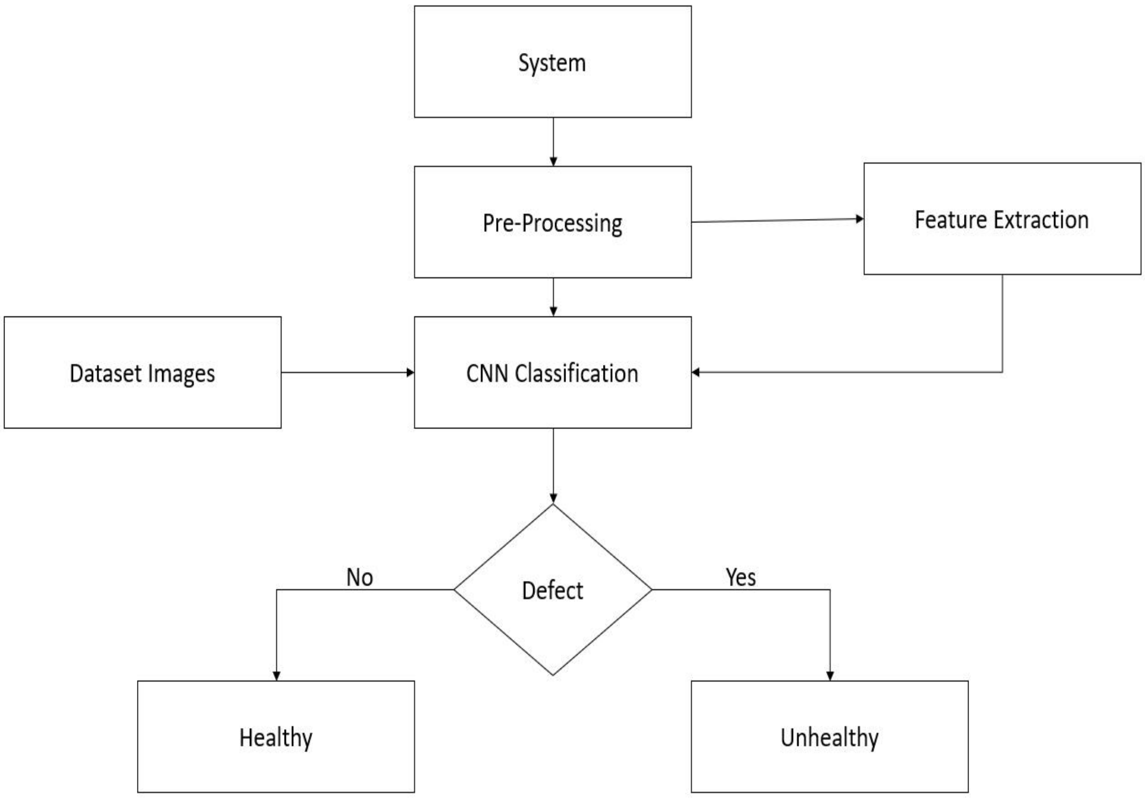


Figure 4.1: Block Diagram of Plant Disease Detection

Dataset Images:

Appropriate data sets are required for all phases of object recognition research, from the training phase to testing the effectiveness of visual algorithms.

The dataset will be classified into diseased leaves and healthy leaves. The dataset will be trained by using a Deep Learning model (Convolutional Neural Network). Therefore, a deep neural network can be trained to separate adjacent leaves.

Diseased Leaves Dataset:

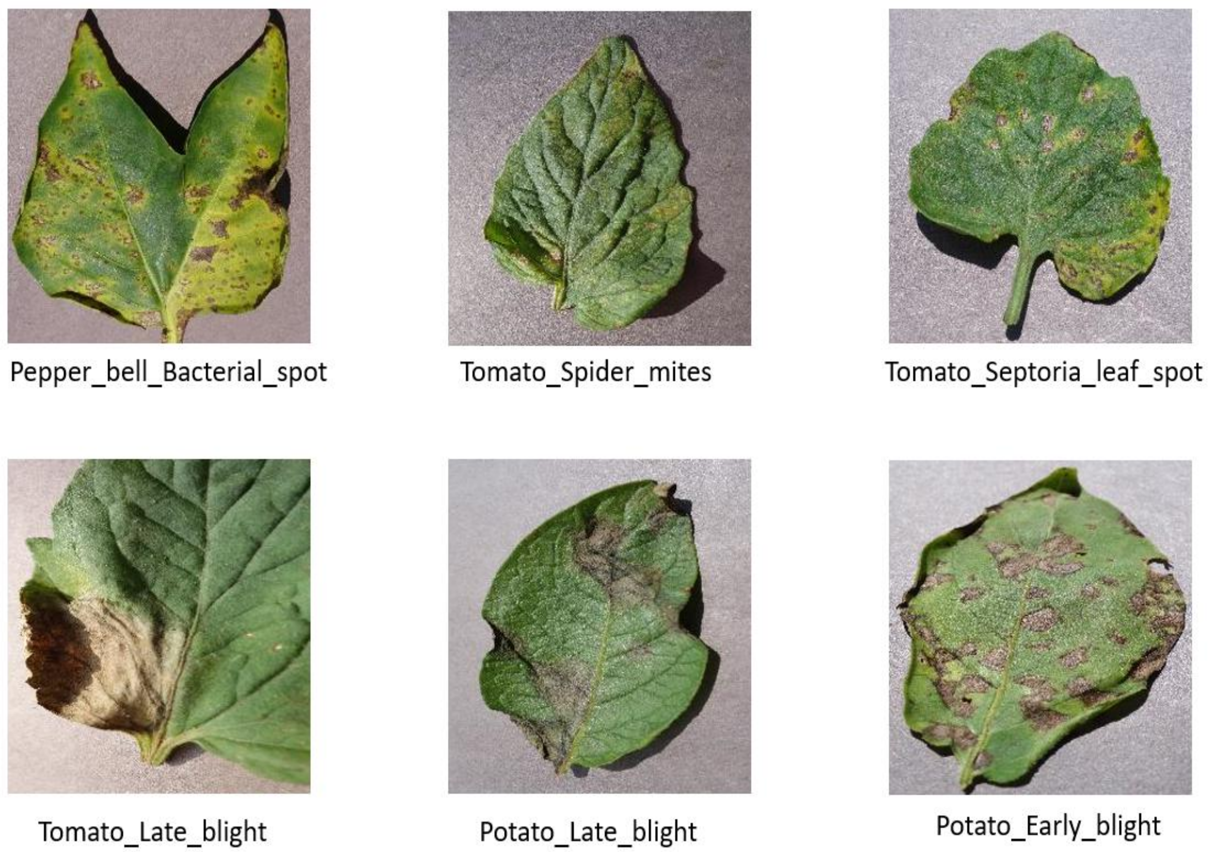


Figure 4.2: Diseased Leaves

Healthy Leaves Dataset:

Figure 4.3: Healthy Leaves

Image Pre-Processing:

In order to get a better feature output, the final images intended for use as a deep neural network configuration database will be processed to determine consistency.

Dataset images were scaled to reduce training time, which was automatically calculated by text in Python, using the Open CV framework.

It is important to use accurately classified images for training and validation data. Only then can a suitable and reliable acquisition model be developed. At this stage, duplicate images that were left after the initial collection and collection of images were removed from the database.

Before sending images to the network, two pre-processing steps are often necessary. First, the images must typically be resized to match the size of the input layer of the CNN. Secondly, the images must be normalized to help the model to converge more quickly as well as to better generalize on unseen data . Even if using color images helps the identification process, as the performance decreases only slightly during the grayscale transformation, this highlights that the network relies mainly on other features to identify diseases. In the same study, the authors also evaluated the impact of background suppression. In fact, background management is one of the challenging elements in the implementation of automatic methods for identifying phytosanitary problems in imagery. With conventional image processing methods, leaf segmentation is a preliminary step to the analysis . The performance obtained by is marginally better with the background, improving the f1-score by slightly <1%. Since background segmentation is not an option on images taken in the field, and since it is the strength of the CNNs to manage complex backgrounds, background suppression is unnecessary.

Feature extraction :

Feature extraction is the process of defining a set of features, or image characteristics, which will most efficiently or meaningfully represent the information that was important for analysis and classification. In this work statistical techniques are used to describe the textures. In this step Grey Level Co-occurrence matrix of the leaf images are calculated. Grey-level co-occurrence matrix (GLCM) creates a matrix from image. Gray Level Co-occurrence Matrix (GLCM) with 256×255 matrices is used in extraction of features, where each matrix position is obtained by applying Scale Invariant Feature Transform (SIFT) in a scaled image.

CNN Classification:

Every Convolutional Neural Network architecture is divided into two parts first is feature extraction and second is classification and has four main components.

1. Convolutional operation. 2. Max-pooling (Down sampling) 3. ReLu (Non Linearity)
4. Classification (fully connected layer)

As soon as image is reached to server it is processed with Algorithm here we extract the feature of image with convolutional operation by convolving the filter over image which produces the feature maps such as edges, texture, spots, holes, colour. These features maps are down sampled so that it can be passed to fully connected layer i.e. classifier after each layer we apply ReLu i.e. non linearity so solve complex problem like classification. Then these maps are flattened and given to fully connected layer where it is classified into the different classes of diseases and return with the name of Disease which has highest probability and corresponding pesticides are send back to the application. Where it is converted into appropriate format and displayed on the screen of user.

Summary:

The dataset is Preprocessed such as Image reshaping ,resizing and conversion to an array form. Similar processing is also done on the test image. A dataset consisting of about 20,638 different plant species is obtained , out of which any image can be used as a test image for the software. The train dataset is used to trainthe model (CNN) so that it can identify the test image and the disease it has .CNN has different layers that are Dense, Dropout, Activation, Flatten,Convolution2D, MaxPooling2D. After the model is trained successfully ,the software can identify the disease if the plant species is contained in the dataset. After successful training and preprocessing ,comparison of the test image and trained model takes place to predict the disease.

CHAPTER 5

RESULTS:

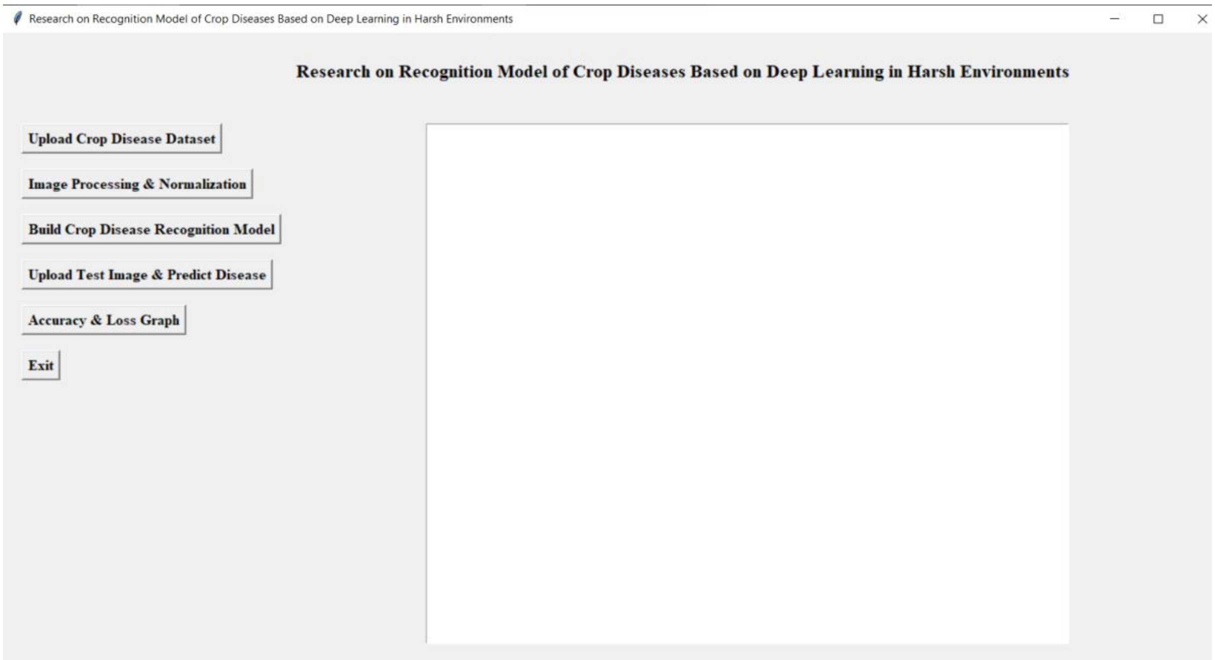


Figure 5.1 : Screen to upload the Dataset

In above screen , click “Upload Crop Disease Dataset”

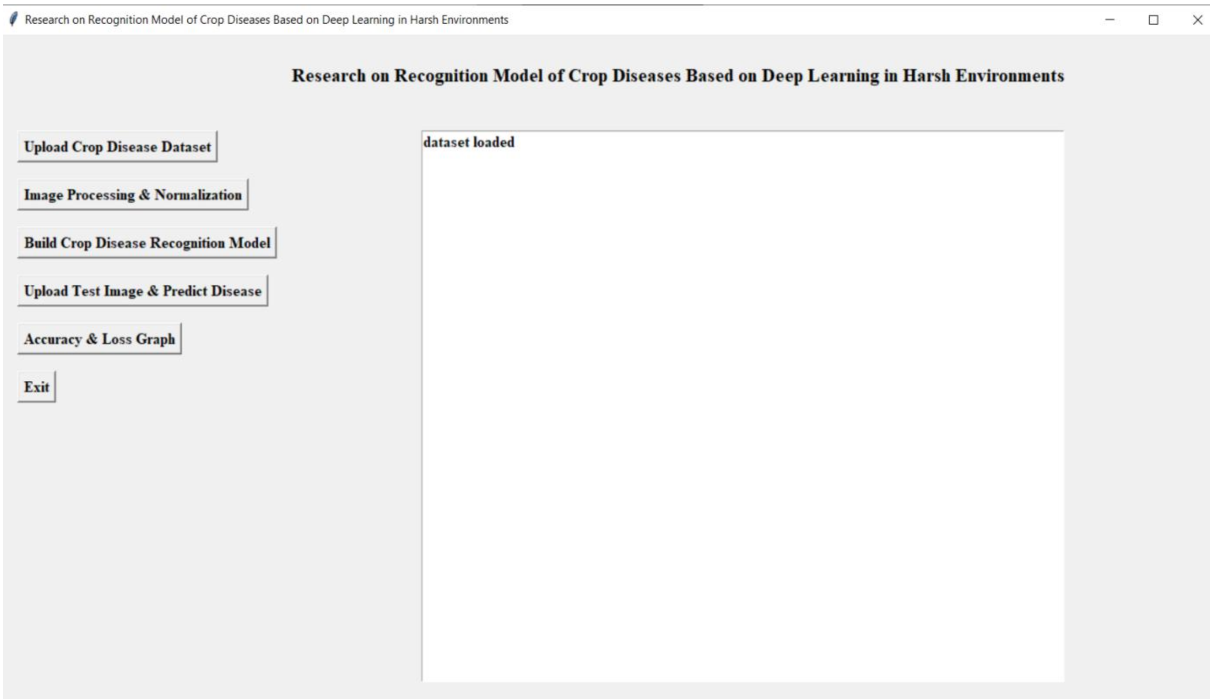


Figure 5.2 : Screen when dataset is uploaded

In above screen dataset loaded and now click on ‘Image Processing & Normalization’ button to read all images and then process images to normalize by converting each image pixel value between 0 and 1 and for that normalization we will divide image pixels with 255 and then get value as 0 or 1 as all images pixel value will be between 0 to 255.

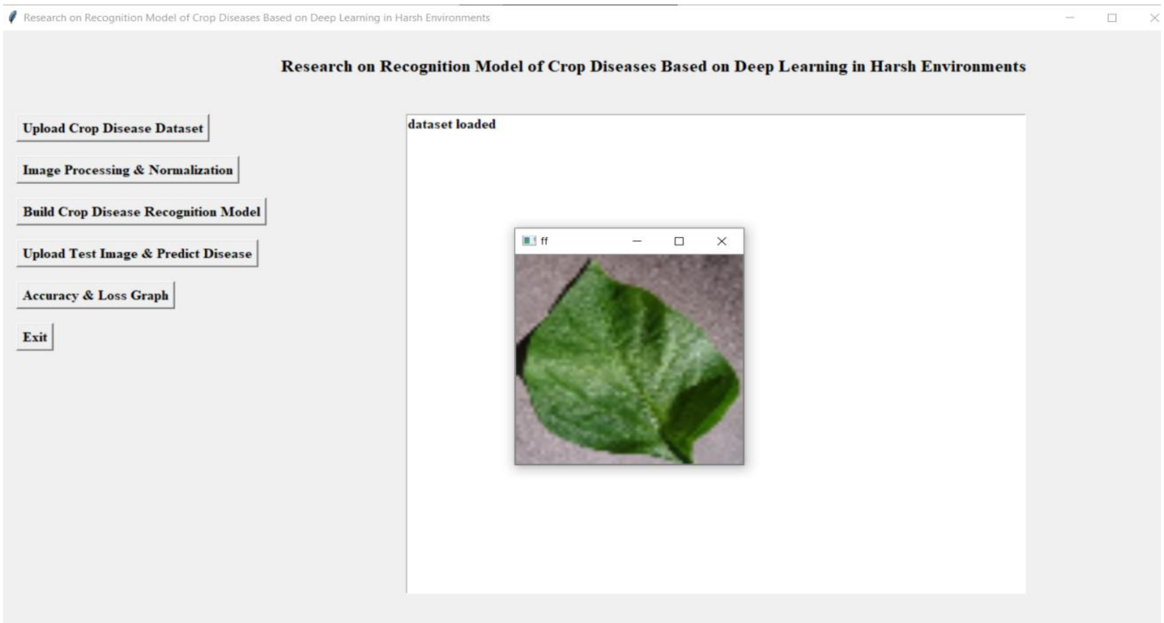


Figure 5.3: Screen that appears while Image processing and Normalisation

In above screen after applying normalization we are just displaying one random image from dataset to check whether images loaded and process properly or not and now you close above image to get below screen

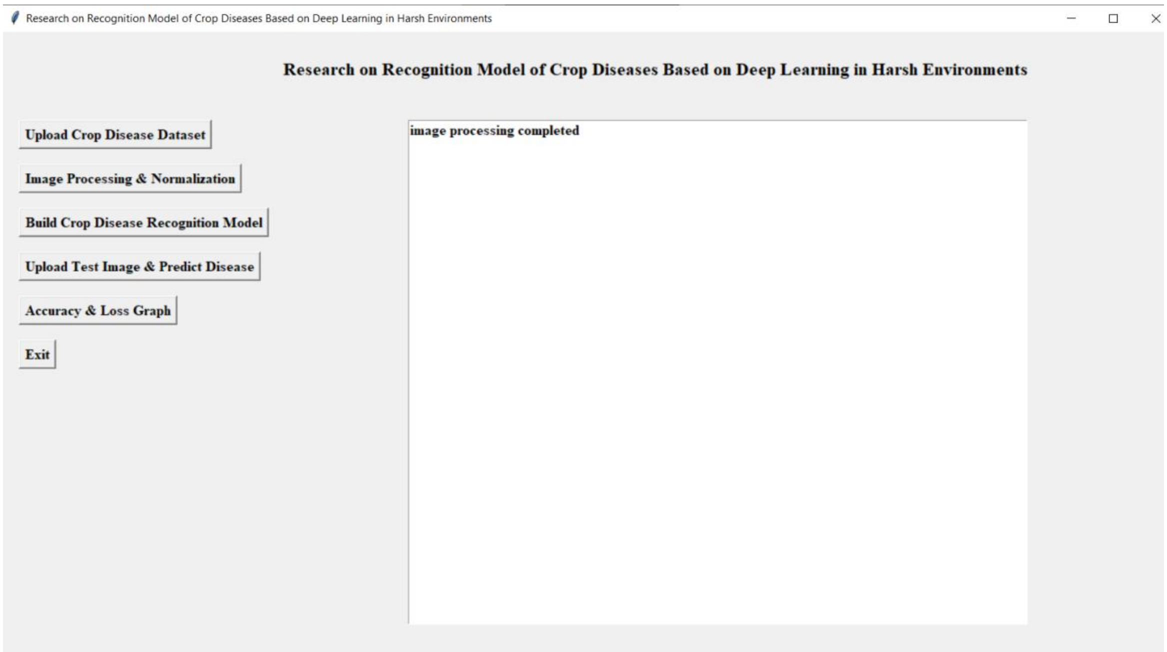


Figure 5.4: Screen that appears after Image processing and Normalisation

In above screen all images processing is done successfully and dataset images are ready and now click on 'Build Crop Disease Recognition Model' button to build CNN model

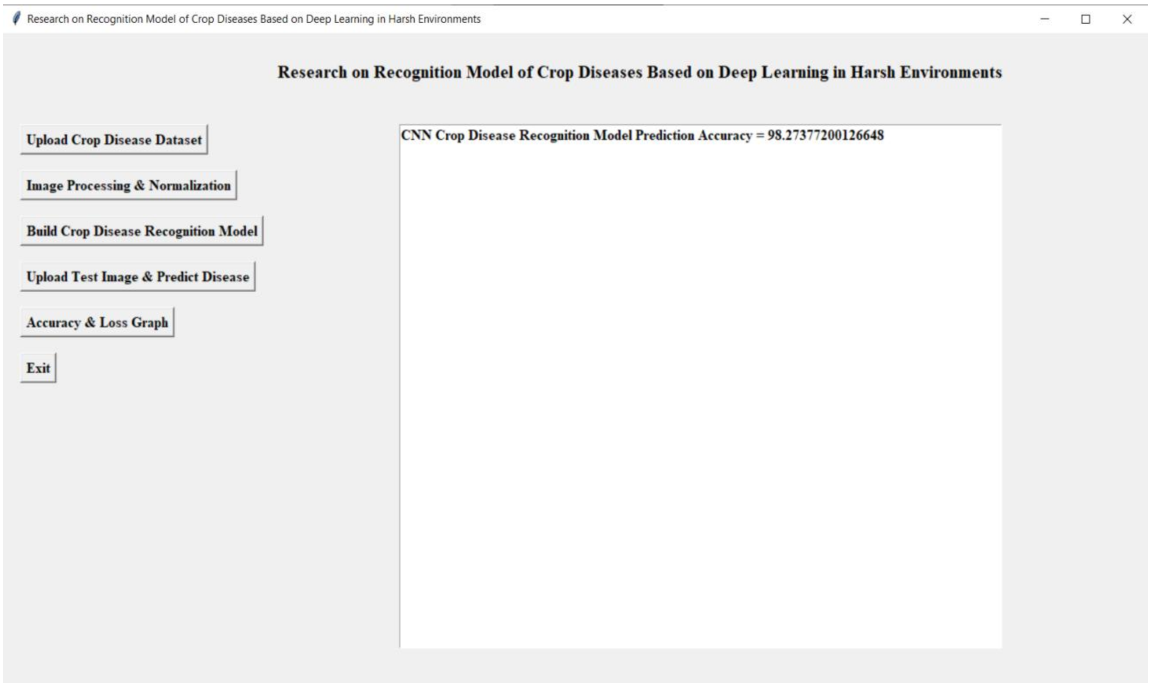


Figure 5.5: Screen that appears after CNN Model generation

In above screen CNN model generated and its overall accuracy is 98% and in below console screen we can see all CNN layers details.

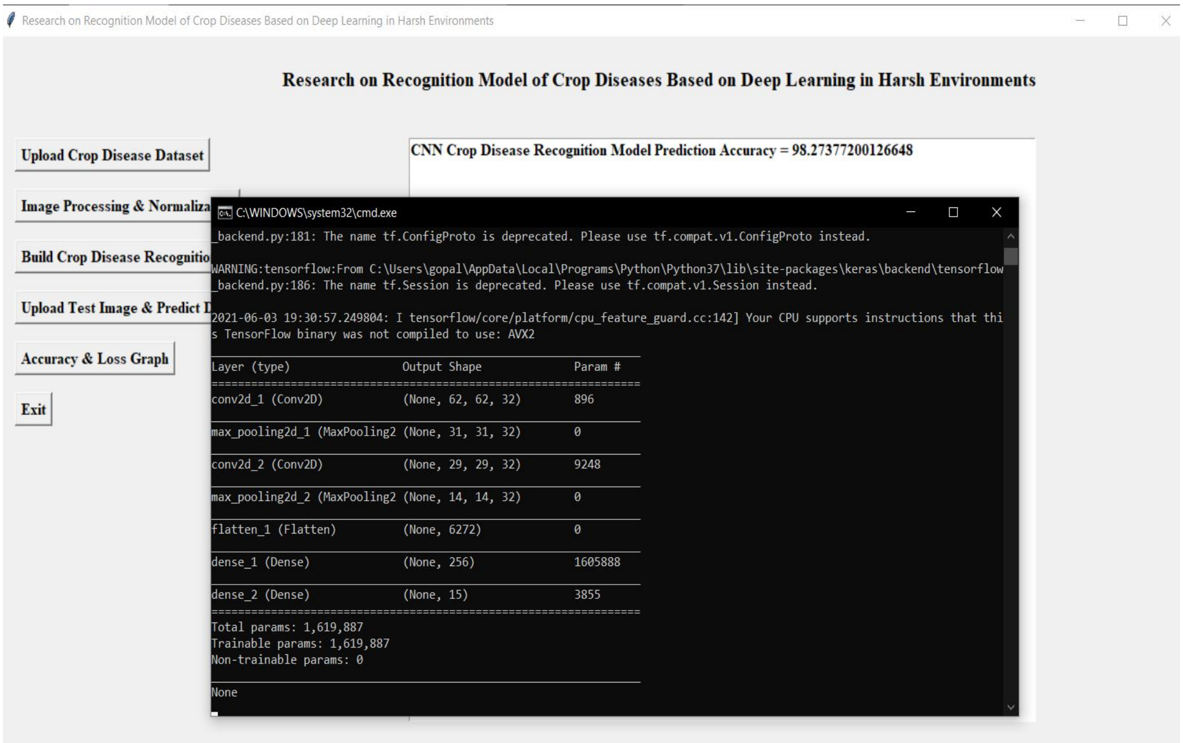


Figure 5.6 : Screen that shows the execution of hidden layers in CNN

In above screen we can see we have used CONV2D, MAXPOOLING, FLATTEN and DENSE layer to build crop disease recognition model and RELU details you can see in code. Now model is ready and now click on ‘Upload Test Image & Predict Disease’ button to upload any test image and then application will predict disease or healthy from that image.

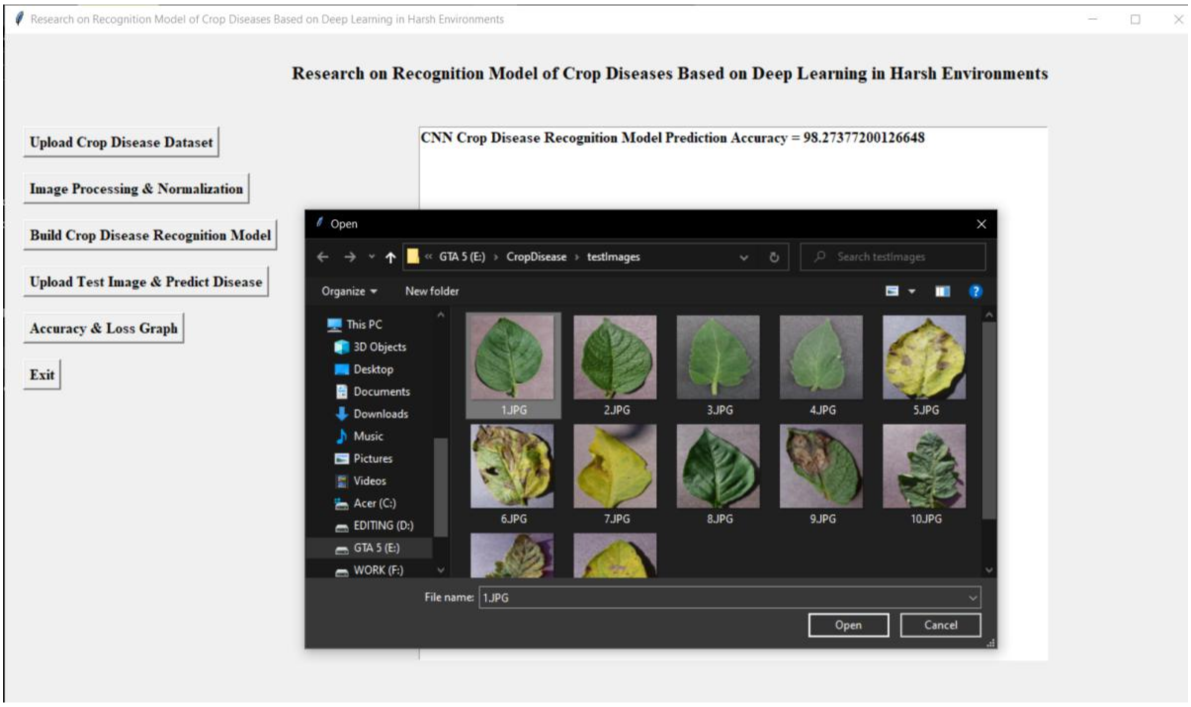


Figure 5.7: Uploading Test image

In above screen selecting and uploading ‘1.JPG’ image file and then click on ‘Open’ button to get below prediction result

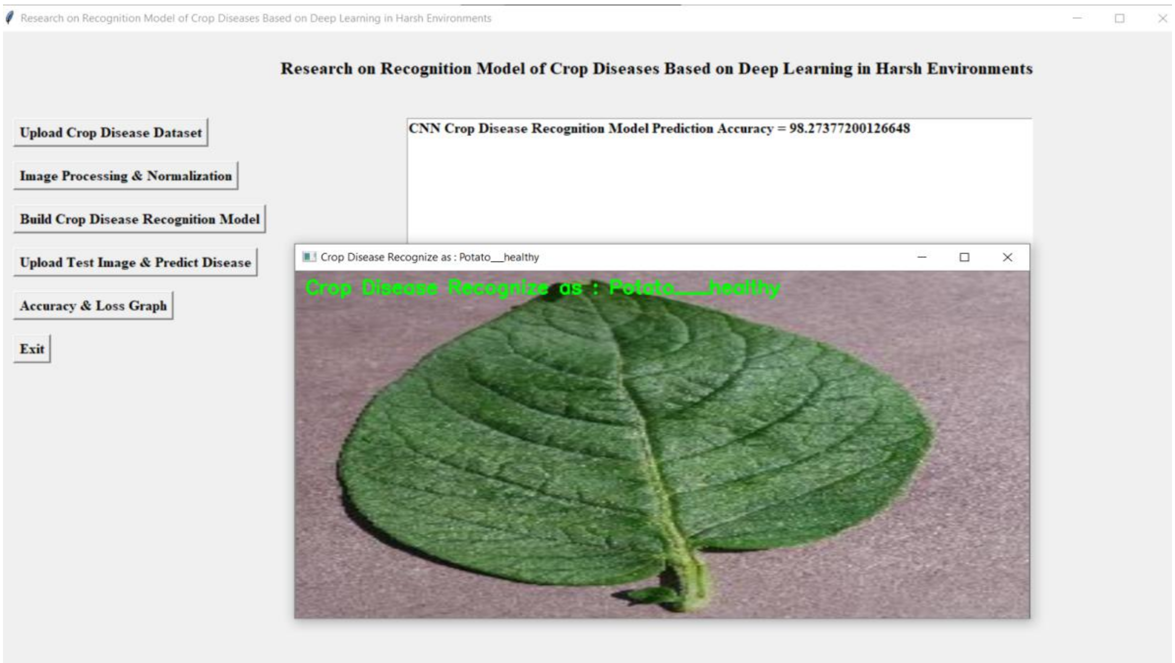


Figure 5.8 : Test Output

In above screen potato leaf predicted as healthy and now try with other image.

Figure 5.9 : Uploading Test image

In above screen selecting and uploading ‘5.JPG’ file and click ‘Open’ button to get below result

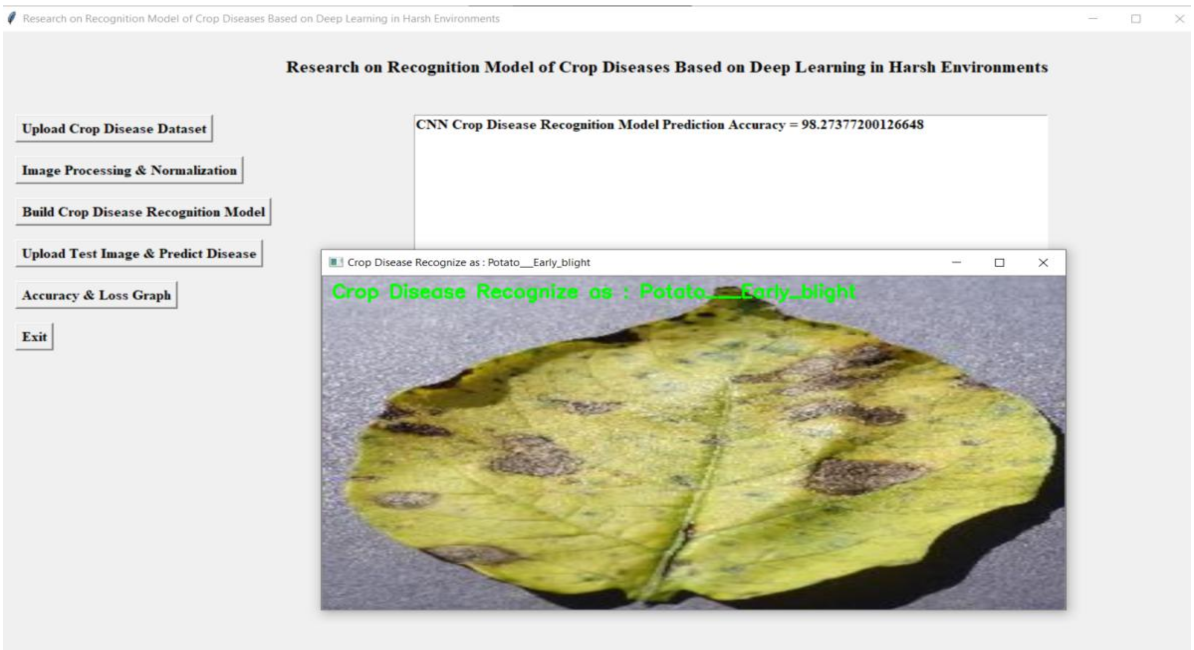


Figure 5.10 : Uploading Test image

In above image potato EARLY BLIGHT disease is detected or recognize and similarly you can upload any other image and get result and now click on ‘Accuracy & Loss Graph’ button to get below graph.

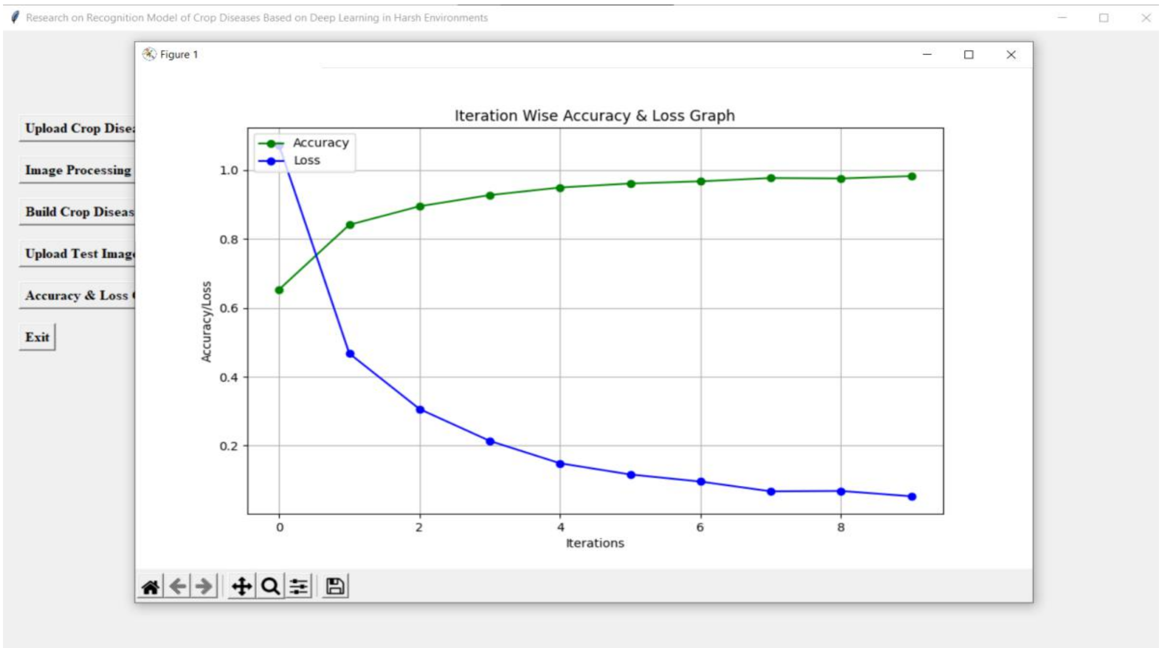


Figure 5.11 : Accuracy & Loss Graph

In above graph x-axis represents epoch/iterations and y-axis represents accuracy/loss and green line represents accuracy and blue line represents loss and from above graph we can see with each increasing iteration accuracy is getting better and better and loss getting decrease.

CHAPTER 6

CONCLUSION AND FUTURE WORK:

6.1.CONCLUSION:

To prevent losses, small holder farmers are dependent on a timely and accurate crop disease diagnosis. In this study, a pre-trained Convolutional Neural Network was fine-tuned, and the model was deployed online. The final result was a plant disease detection app. This service is free, easy to use and requires just a smart phone and internet connection. Thus, the user’s needs as defined in this paper have been fulfilled.

We proposed Crop Disease Detection using CNN system based on Deep Learning. The described system can be efficiently used by farmers as it is gives the instant information about the crop disease. It also reduces the Outbreaks, upsurges which causes the huge losses to crops and pastures and threatening the livelihoods of vulnerable farmers. As Comparing with traditional crop disease detection system, the described system gives the accuracy rate of 98% which implies correct detection. The experimental results demonstrate the effectiveness of our proposed system and it can be used widely by Farmers to Detect the crop Disease.

Overall, this study is conclusive in demonstrating how CNNs may be applied to empower small-holder farmers in their fight against plant disease. In the future, work should be focused on diversifying training datasets and also in testing similar web applications in real life situations. Without such developments, the struggle against plant disease will continue.

6.2. FUTURE WORK:

This system considers only the leaf of the plant to detect the disease of that crop. It will be more convenient if the other parts of the crop such as roots, stem, branches etc. which increases the detection accuracy more than current one. Also image categorization will also be done to check whether the given leaf is of preferred category or not. If a model provided with input other than leaf image then also it shows some name of disease for it.

REFERENCES

1. Chouhan, Siddharth Singh, Ajay Kaul, Uday Pratap Singh, and Sanjeev Jain. "Bacterial foraging optimization based radial basis function neural network (BRBFNN) for identification and classification of plant leaf diseases: An automatic approach towards plant pathology." IEEE Access 6 (2018): 8852-8863.
2. Fadzil, WM N. Wan Mohd, MS B. Shah Rizam, R.Jailani, and M. T. Nooritawati. "Orchid leaf disease detection using border segmentation techniques." In 2014 IEEE Conference on Systems, Process and Control (ICSPC 2014), pp. 168-173. IEEE, 2014.
3. Parikh, Aditya, Mehul S. Raval, Chandrasinh Parmar, and Sanjay Chaudhary. "Disease detection and severity estimation in cotton plant from unconstrained images." In 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), pp. 594-601. IEEE, 2016.]
4. Prajapati, Bhumika S., Vipul K. Dabhi, and Harshadkumar B. Prajapati. "A survey on detection and classification of cotton leaf diseases." In 2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT), pp. 2499-2506. IEEE, 2016.
5. Rothe, P. R., and R. V. Kshirsagar. "Cotton leaf disease identification using pattern recognition techniques." In 2015 International Conference on Pervasive Computing (ICPC), pp. 1-6. IEEE, 2015.
6. Sardogan, Melike, Adem Tuncer, and Yunus Ozen. "Plant leaf disease detection and classification based on CNN with LVQ algorithm." In 2018 3rd International Conference on Computer Science and Engineering (UBMK), pp. 382-385. IEEE, 2018.

7. Yusoff, Norfarahin Mohd, Ili Shairah Abdul Halim, and Noor Ezan Abdullah. "Real-time Hevea Leaves Diseases Identification using Sobel Edge Algorithm on FPGA: A Preliminary Study." In 2018 9th IEEE Control and System Graduate Research Colloquium (ICSGRC), pp. 168-171. IEEE, 2018.

8. Indumathi, R., N. Saagari, V. Thejuswini, and R. Swarnareka. "Leaf Disease Detection and Fertilizer Suggestion." In 2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN), pp. 1-7. IEEE, 2019.

9. Kuricheti, Gayatri, and P. Supriya. "Computer Vision Based Turmeric Leaf Disease Detection and Classification: A Step to Smart Agriculture." In 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI), pp. 545-549. IEEE, 2019.

10. Khitthuk, Chaowalit, Arthit Srikaew, Kitti Attakitmongcol, and Prayoth Kumsawat. "Plant Leaf Disease Diagnosis from Color Imagery Using Co-Occurrence Matrix and Artificial Intelligence System." In 2018 International Electrical Engineering Congress (iEECON), pp. 1-4. IEEE, 2018.

11. Jiang, Peng, Yuehan Chen, Bin Liu, Dongjian He, and Chunquan Liang. "Real-Time Detection of Apple Leaf Diseases Using Deep Learning Approach Based on Improved Convolutional Neural Networks." IEEE Access 7 (2019): 59069-59080.

APPENDIX:

Software Code:

```
from tkinter import messagebox
from tkinter import *
from tkinter import simpledialog
import tkinter
import matplotlib.pyplot as plt
import numpy as np
from tkinter import ttk
from tkinter import filedialog
from keras.utils.np_utils import to_categorical
from keras.models import Sequential
from keras.layers.core import Dense,Activation,Dropout, Flatten
from sklearn.metrics import accuracy_score
import os
import cv2
from keras.layers import Convolution2D
from keras.layers import MaxPooling2D
import pickle
from keras.models import model_from_json
```

```
main = Tk()
main.title("Research on Recognition Model of Crop Diseases Based on Deep Learning
in Harsh Environments")
main.geometry("1300x1200")

global filename
global X, Y
global model
global accuracy

plants = ['Pepper_bell_Bacterial_spot', 'Pepperbell_healthy', 'Potato_Early_blight',
'Potato_healthy', 'Potato_Late_blight', 'Tomato_Target_Spot',
        'Tomato_Tomato_mosaic_virus', 'TomatoTomato_YellowLeaf_Curl_Virus',
'Tomato_Bacterial_spot', 'Tomato_Early_blight', 'Tomato_healthy',
        'Tomato_Late_blight', 'Tomato_Leaf_Mold', 'Tomato_Septoria_leaf_spot',
'Tomato_Spider_mites_Two_spotted_spider_mite']

def uploadDataset():
   global X, Y
   global filename
   text.delete('1.0', END)
   filename = filedialog.askdirectory(initialdir=".")
   text.insert(END,'dataset loaded\n')

def imageProcessing():
   text.delete('1.0', END)
   global X, Y
   X = np.load("model/myimg_data.txt.npy")
   Y = np.load("model/myimg_label.txt.npy")
   Y = to_categorical(Y)
   X = np.asarray(X)
   Y = np.asarray(Y)
   X = X.astype('float32')
   X = X/255
   indices = np.arange(X.shape[0])
   np.random.shuffle(indices)
   X = X[indices]
   Y = Y[indices]
   text.insert(END,'image processing completed\n')
   img = X[20].reshape(64,64,3)
   cv2.imshow('ff',cv2.resize(img,(250,250)))
   cv2.waitKey(0)

def cnnModel():
   global model
   global accuracy
   text.delete('1.0', END)
   if os.path.exists('model/model.json'):
      with open('model/model.json', "r") as json_file:
```

```
            loaded_model_json = json_file.read()
            model = model_from_json(loaded_model_json)
        json_file.close()
        model.load_weights("model/model_weights.h5")
        model._make_predict_function()
        print(model.summary())
        f = open('model/history.pckl', 'rb')
        accuracy = pickle.load(f)
        f.close()
        acc = accuracy['accuracy']
        acc = acc[9] * 100
        text.insert(END,"CNN Crop Disease Recognition Model Prediction Accuracy = 
"+str(acc))
    else:
        model = Sequential() #resnet transfer learning code here
        model.add(Convolution2D(32, 3, 3, input_shape = (64, 64, 3), activation = 'relu'))
        model.add(MaxPooling2D(pool_size = (2, 2)))
        model.add(Convolution2D(32, 3, 3, activation = 'relu'))
        model.add(MaxPooling2D(pool_size = (2, 2)))
        model.add(Flatten())
        model.add(Dense(output_dim = 256, activation = 'relu'))
        model.add(Dense(output_dim = 15, activation = 'softmax'))
        model.compile(optimizer = 'adam', loss = 'categorical_crossentropy', metrics = 
['accuracy'])
        print(model.summary())
        hist = model.fit(X, Y, batch_size=16, epochs=10, validation_split=0.2, 
shuffle=True, verbose=2)
        model.save_weights('model/model_weights.h5')
        model_json = model.to_json()
        with open("model/model.json", "w") as json_file:
            json_file.write(model_json)
        json_file.close()
        f = open('model/history.pckl', 'wb')
        pickle.dump(hist.history, f)
        f.close()
        f = open('model/history.pckl', 'rb')
        accuracy = pickle.load(f)
        f.close()
        acc = accuracy['accuracy']
        acc = acc[9] * 100
        text.insert(END,"CNN Crop Disease Recognition Model Prediction Accuracy = 
"+str(acc))

def predict():
    global model
    filename = filedialog.askopenfilename(initialdir="testImages")
    img = cv2.imread(filename)
    img = cv2.resize(img, (64,64))
    im2arr = np.array(img)
    im2arr = im2arr.reshape(1,64,64,3)
```

```
    test = np.asarray(im2arr)
    test = test.astype('float32')
    test = test/255
    preds = model.predict(test)
    predict = np.argmax(preds)
    img = cv2.imread(filename)
    img = cv2.resize(img, (800,400))
    cv2.putText(img, 'Crop Disease Recognize as : '+plants[predict], (10, 25),
cv2.FONT_HERSHEY_SIMPLEX,0.7, (0, 255, 0), 2)
    cv2.imshow('Crop Disease Recognize as : '+plants[predict], img)
    cv2.waitKey(0)

def graph():
    acc = accuracy['accuracy']
    loss = accuracy['loss']
    plt.figure(figsize=(10,6))
    plt.grid(True)
    plt.xlabel('Iterations')
    plt.ylabel('Accuracy/Loss')
    plt.plot(acc, 'ro-', color = 'green')
    plt.plot(loss, 'ro-', color = 'blue')
    plt.legend(['Accuracy', 'Loss'], loc='upper left')
    #plt.xticks(wordloss.index)
    plt.title('Iteration Wise Accuracy & Loss Graph')
    plt.show()

def close():
    main.destroy()
    text.delete('1.0', END)

font = ('times', 15, 'bold')
title = Label(main, text='Research on Recognition Model of Crop Diseases Based on
Deep Learning in Harsh Environments')
#title.config(bg='powder blue', fg='olive drab')
title.config(font=font)
title.config(height=3, width=120)
title.place(x=0,y=5)

font1 = ('times', 13, 'bold')
ff = ('times', 12, 'bold')

uploadButton = Button(main, text="Upload Crop Disease Dataset",
command=uploadDataset)
uploadButton.place(x=20,y=100)
uploadButton.config(font=ff)


processButton = Button(main, text="Image Processing & Normalization",
command=imageProcessing)
processButton.place(x=20,y=150)
```

```
processButton.config(font=ff)

modelButton = Button(main, text="Build Crop Disease Recognition Model",
command=cnnModel)
modelButton.place(x=20,y=200)
modelButton.config(font=ff)

predictButton = Button(main, text="Upload Test Image & Predict Disease",
command=predict)
predictButton.place(x=20,y=250)
predictButton.config(font=ff)

graphButton = Button(main, text="Accuracy & Loss Graph", command=graph)
graphButton.place(x=20,y=300)
graphButton.config(font=ff)

exitButton = Button(main, text="Exit", command=close)
exitButton.place(x=20,y=350)
exitButton.config(font=ff)

font1 = ('times', 12, 'bold')
text=Text(main,height=30,width=85)
scroll=Scrollbar(text)
text.configure(yscrollcommand=scroll.set)
text.place(x=450,y=100)
text.config(font=font1)

main.config()
main.mainloop()
```