PLANT LEAF DISEASE DETECTION AND CLASSIFICATION USING ANN

A Mini Project Report submitted to
JNTU Hyderabad in partial fulfilment
of the requirements for the award of the degree

BACHELOR OF TECHNOLOGY

In

ELECTRONICS AND COMMUNICATION ENGINEERING

Submitted by

BATTULA SHILPA	**20RG1A0468**
BOBIDE VAISHNAVI	**20RG1A0470**
KOMPALLY TANUJA	**20RG1A0487**
KANKARLA YASHASWI	**21RG5A0406**

Under the Guidance of
Mrs. A. JHANSI RANI
B. Tech., M. Tech., (Ph.D.)
Assistant Professor

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

MALLA REDDY COLLEGE OF ENGINEERING FOR WOMEN
(Approved by AICTE New Delhi and Affiliated to JNTUH)
(An ISO 9001: 2015 Certified Institution)
(B.Tech Programs(CSE,ECE) Accredited by NBA)
Maisammaguda,Medchal (M), Hyderabad-500100, T.S

SEPTEMBER 2023

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

MALLA REDDY COLLEGE OF ENGINEERING FOR WOMEN

(Approved by AICTE New Delhi and Affiliated to JNTUH)
(An ISO 9001: 2015 Certified Institution)
(B.Tech Programs(CSE,ECE) Accredited by NBA)
Maisammaguda, Medchal (M), Hyderabad-500100, T. S.
SEPTEMBER 2023

CERTIFICATE

This is to certify that the Mini project entitled **"PLANT LEAF DISEASE DETECTION AND CLASSIFICATION USING ANN"** has been submitted by **Battula Shilpa (20RG1A0468), Bobide Vaishnavi (20RG1A0470), Kompally Tanuja (20RG1A0487), Kankarla Yashaswi (21RG5A0406)** in partial fulfillment of the requirements for the award of **BACHELOR OF TECHNOLOGY** in **ELECTRONICS & COMMUNICATION ENGINEERING**. This record of bonafide work was carried out by them under my guidance and supervision. ***The result embodied in this mini-project report has not been submitted to any other University or Institute for the award of any degree.***

Mrs. A. Jhansi Rani
Assistant Professor
Project Guide

Dr. Selvamani Indrajith
Head of the Department

External Examiner

ACKNOWLEDGEMENT

The Mini Project work carried out by our team in the Department of Electronics and Communication Engineering, Malla Reddy College of Engineering for Women, Hyderabad. ***This work is original and has not been submitted in part or full for any degree or diploma of any other university.***

We wish to acknowledge our sincere thanks to our project guide ***Mrs.A. Jhansi Rani***, Assistant Professor of Electronics & Communication Engineering Department for formulating the problem, analysis, guidance and her continuous supervision during the project work.

We acknowledge our sincere thanks to ***Dr. Vaka Murali Mohan***, Director, MRCEW. ***Dr. Kanaka Durga Returi,*** Principal and Professor of Computer Science & Engineering Department. ***Dr. Selvamani Indrajith***, Professor and Head of the Electronics & Communication Engineering Department, for their kind cooperation in making this Mini Project work a success.

We extend our gratitude to ***Sri. Ch. Malla Reddy***, Founder-Chairman and ***Sri. Ch. Mahender Reddy,*** Secretary, MRCEW and ***Sri. P. Praveen Reddy*** Director, MRGI for their kind cooperation in providing the infrastructure for successful completion of our Mini project work.

We convey our special thanks to the Entire Teaching faculty and non-teaching staff members of the Electronics & Communication Engineering Department for their support in making this project work a success.

BATTULA SHILPA	**(20RG1A0468)**	___________
BOBIDE VAISHNAVI	**(20RG1A0470)**	___________
KOMPALLY TANUJA	**(20RG1A0487)**	___________
KANKARLA YASHASWI	**(20RG5A0406)**	___________

INDEX

ABSTRACT

Nowadays, herb plants are important for humans as well as for the medical field. In this report, the Artificial Neural Network is used, to analyze and classify whether the leaf is healthy or unhealthy. In the agriculture field, technology is used in classification, image processing, and image acquisition for the analysis quality of leaves. For the color transformation structure, RGB to HSV transformation is applied to the image, the most important thing for this part is the segmentation for the method of image processing. On the basis of shape and color, the analysis of the leaf disease image is performed. Finally, in the classification method, we used the back-propagation algorithm on the feed-forward neural network. The classifier of neural networks shows better performance and more accuracy.

LIST OF FIGURES

LIST OF TABLES

CHAPTER1: INTRODUCTION

1.1Project Overview:

In this research, the aim is to develop a technology in the agriculture field, based on engineering techniques. Nowadays, crops face many diseases. Damage caused by the insect is one of the major traits/diseases. Insecticides are not always proven to be efficient because insecticides may be toxic to some kind of birds.

Method used for this project is to get leaf image from herb plant. Method of extraction the image acquisition uses suitable image processing algorithms and then makes recognition and classification of healthy or unhealthy leaves using Artificial Neural Network. Image pattern of classification for this project is based on color and area of leaf

Machine learning tasks are broadly classified into supervised learning, unsupervised learning, reinforcement learning, Semi-supervised learning and learning models (classification, regression, clustering, and dimensionality reduction). Table 1.1 describes difference between supervised learning and unsupervised learning.

Table 1.1 Difference between supervised learning and unsupervised learning.

FACTORS	SUPERVISED LEARNING	UNSUPERVISED LEARNING
Input	Known and labeled data	Unknown data
Complexity	Very complex	Less Complex
Number of classes	Known	Unknown
Accuracy	Accurate and reliable	Moderately Accurate and reliable

Machine Learning Algorithm

- Bayesian belief network
- Black-propagation network
- Classification and regression trees
- Chi-square automatic interaction detector
- Convolutional neural networks
- Counter propagation
- Deep belief network
- Deep neural networks
- Extreme learning machines
- Expectation maximization
- Ensemble neural network
- Gaussian naive Bayes
- Generalized regression neural network
- Machine Learning Algorithm
- Successive projection algorithm-support vector machine
- K-nearest neighbor
- Linear discriminant analysis
- Least squares-support vector machine
- Multivariate adaptive regression splines
- Multi-layer perceptron
- Multiple Linear Regressions
- Principal component analysis
- Random forest
- Support vector regression
- Partial least square regression

Figure 1.1 Machine learning algorithms

1.2 Machine Learning Methods

There are various Machine learning methods such as boosting techniques (RGF, GBDT, and Ada boost), Regression Tree (ID3, C4.5), Random Forest, SVM, K Nearest Neighbor, ANN, etc.

1.3 Artificial Neural Network

Artificial Neural Network is a simple mathematical model of the brain. This is used to process nonlinear relationships between inputs and outputs in parallel form, for example human brain. So Artificial Neural Networks can be used in different variety of tasks one of the best use is classification. We can learn Artificial Neural Network speedily. The Information flows through a neural network in two different ways.

Also, each of the neurons has a number and a special formula associated with it called as threshold value. The neural network can be trained and then it can be provided with a set of inputs and outputs. Each neuron transforms the input and forwards it to the next layer and so on. The result is received on the output layer.

Each neuron receives inputs to its left, and then they are multiplied by the weights. So every neuron adds up all the inputs. If the sum is more than a certain threshold value then the neuron "fires". Whenever we use large datasets, the neural networks are more powerful at that time.

1.4 Simple Artificial Neural Network

SVM stands for Support Vector Machine. It is a simple algorithm in machine learning. It belongs to the supervised learning category in

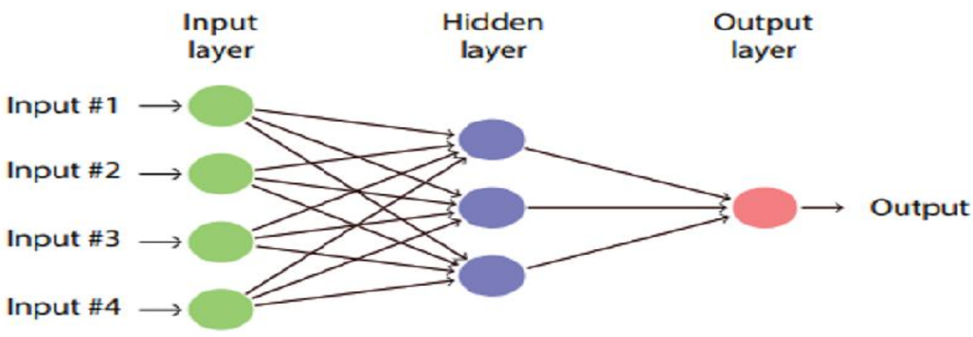


Figure1.4 Simple Artificial Neural Network

The analysis of the data related to this field helps identify how the use of the latest technology can be improved. The images of leaves and other parts of the plants can be used to detect diseases in plants. Visualization techniques have also been included in this technology, and it has been improved over the last three years to the current improved levels.

The challenges that face the world today, related to the diseases affecting plants and humans, can be reduced if the diseases are identified before they spread to vast areas. The use of ML is widespread in the world today. Diverse methods used in ML and DL help the experts to analyze the plant diseases and know their source in time.

The importance of plants in the world has increased over time. The discoveries about the critical roles that plants could play in medicine, energy production, and the recent concerns about the reduction of global warming have for long been a significant part of science and technology. A reduction in the plant cover in the world increases the risk of higher global warming and an increase in the related challenges.

The background study shows that the scanning of the images that show the healthy and unhealthy plants forms a basis for comparison by the scientists in this field. DL can be used to detect abnormalities in both humans and plants. The pixel-wise operations are used to analyze the leaves collected from sick plants, and this is used to classify the diseases according to their impact on the plants.

The images are converted into data that the machines and computers can interpret. It means that the technology requires the coding of the images into data that can be fed into the computer systems. Therefore, the basic knowledge in computer codes and programming forms an integral part of the background knowledge required in this field.

Adenocarcinoma of the prostate is one of the most common forms of cancer. It is detected using the image-based detection methods whereby the body is scanned for the abnormalities and the images obtained are used to determine if the patient has the disease.

The images of the leaves can be analyzed using the image-based detection system to determine the classification that suits the disease. In places like sub-Saharan Africa, Cassava is one of the essential foods because it provides people with many carbohydrates. However, due to its vulnerability to viral diseases, it has not effectively sustained the nutritional value it could achieve in this region.

The other challenge is that the symptoms and characteristics of the diseases are diverse and could be similar to a certain degree. For example, many diseases could lead to the wilting of leaves. The challenge is yet to be resolved because more and new images are uploaded progressively by experts. The other challenge is the lack of suitable instruments for use in the work of image detection.

The challenge of inadequate information about the diseases can be solved if there is an improved data-captioning process that involves fine details of the images taken and the differences that define them. The images should be analyzed keenly to determine the ones that are affected or infected. Another solution would be to focus on using the latest technology that is reliable and valid.

CHAPTER 2: LITERATURE SURVEY

C.V. Rubesh [1] stated that "Plant disease automation in agriculture science is the primary concern for every country, as the food demand is increasing at a fast rate due to an increase in population. Moreover, the increased use of technology today has increased the efficacy and accuracy of detecting diseases in plants and animals.

Savary, Serge [2] stated that "The global burden of pathogens and pests on major food crops. Nature ecology & evolution Crop pathogens and pests reduce the yield and quality of agricultural production. They cause substantial economic losses and reduce food security at household, national and global levels.

A. Ramcharan, P. McCloskey, K. Baranowski et al [3] explained that "A Mobile- based Deep Learning Model for Cassava Disease Diagnosis. The article describes a mobile-based deep learning model developed for the diagnosis of cassava diseases. Cassava is an important staple crop in many regions of the world, and its production is threatened by various diseases.

K. Nagpal, D. Foote, Y. Liu et al [4] proposed that "development and validation of a deep learning algorithm for improving Gleason scoring of prostate cancer, described the development and validation of a deep learning algorithm for improving the accuracy of Gleason scoring of prostate cancer.

S. Verma, A. Chug, A. P. Singh, S. Sharma, and P. Rajvanshi [5] explained that "Deep Learning-Based Mobile Application for Plant Disease Diagnosis: A Proof of Concept with a Case Study on Tomato Plant, describes a proof-of-concept mobile application that uses deep learning techniques for the diagnosis of plant diseases.

T. Fujith, E., et al [6] explained that "A practical plant diagnosis system for field leaf images and feature visualization. An accurate, fast and low-cost automated plant diagnosis plant diagnosis system has been called for While several studies utilizing machine learning techniques have been conducted.

A. Kamilaris and F. X. Prenafeta-Boldu [7] elucidated that "Deep learning in agriculture: a survey, it provides a comprehensive survey of the application of deep learning techniques in agriculture. The current state of the art of deep learning in agriculture, including its applications in crop yield prediction, plant disease diagnosis, irrigation management, and soil analysis.

K. P. Ferentinos [8] explained that "Deep learning models for plant disease detection and diagnosis, it focuses on the application of deep learning models for plant disease detection and diagnosis. Provides a comprehensive review of the current state of the art in this field, including the different types of deep learning models used, such as Artificial neural networks (ANNs) and recurrent neural networks (RNNs).

J. G. A. Barbedo [9] stated that "Factors influencing the use of deep learning for plant disease recognition, It discusses the different types of data used for training and validation, such as hyperspectral imagery, RGB imagery, and sensor data. The advantages and disadvantages of each type of data and provides insights into the challenges and opportunities associated with the use of deep learning for plant disease recognition.

A. Dhakal and S. Shakya [10] explained that "Image-based plant disease detection with deep learning, focuses on the use of deep learning models for image-based plant disease detection. A comprehensive review of the literature on this topic, including the different types of deep learning models used, such as convolutional neural networks (CNNs) and transfer learning models.

S. Wallelign, M. Polceanu, and C. Buche [11] stated that "Soybean plant disease identification using convolutional neural network focuses on the use of convolutional neural networks (CNNs) for the identification of soybean plant diseases. The authors provide a comprehensive review of the literature on this topic, including the different types of CNN architectures used and the different datasets used for training and validation.

T. Wiesner-Hanks, E. L. Stewart, N. Kaczmar et al [12] explained that "Image set for deep learning: field images of maize annotated with disease symptoms, The article focuses on the development of a dataset of annotated images of maize plants affected by different diseases for use in deep learning models.

G. Wang, Y. Sun, and J. Wang [13] stated that "Automatic image-based plant disease severity estimation using deep learning, the method for automatically estimating the severity of plant diseases using deep learning. The authors collected a dataset of images of tomato plants affected by different diseases, with varying degrees of severity.

A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes [14] explained that "Deep learning for image-based cassava disease detection, The potential applications of their method in the field of plant disease diagnosis and management. The method presented in the article has the potential to help farmers to make more informed decisions.

Mohanty, Sharada P., David P. Hughes, and Marcel Salathé [15] stated "Using deep learning for image-based plant disease detection. Crop diseases are a major threat to food security, but their rapid identification remain difficult in many parts of the world due to the lack of the necessary infrastructure.

Szegedy, Christian, et al [16] explained that "Going deeper with convolutions, we propose a deep convolutional neural network architecture codenamed Inception that achieves the new state of the art for classification and detection in the Image Net Large-Scale Visual Recognition Challenge 2014 (ILSVRC14).

Deng, Jia, et al [17] states that "ImageNet: A large-scale hierarchical image. It describes the ImageNet project, which is a large-scale image database with over 14 million images that are categorized into more than 20,000 classes. The creation of this database and the challenges involved in developing a large-scale dataset for computer vision research.

Cunningham, Padraig, and Sarah Jane Delany [18] stated that "K- Nearest neighbor classifiers. K-Nearest Neighbors is one of the simplest Machine Learning algorithms based on Supervised Learning technique. K-NN algorithm assumes the similarity between the new case/data and available cases.

Duan, Kai-Bo, and S. Sathya Keerthi [19] stated that "Which is the best multiclass SVM method? An empirical study. binary (two- class) classification using support vector machines (SVMs) is a very well-developed technique.

Cortes, Corinna, and Vladimir Vapnik [20] explained that "Support-vector networks. Machine Learning 20.3 The support-vector network is a new learning machine for two-group classification problems. The machine conceptually implements the following idea: input vectors are non-linearly mapped to a very high-dimensional feature space. In this feature space a linear decision surface is constructed.

CHAPTER 3: EXISTING SYSTEM

Agriculture is an important source in the economic development of India.

3.1 Machine Learning Methods:

There are various Machine learning methods such as boosting techniques (RGF, GBDT, and Ada boost), Regression Tree (ID3, C4.5), Random Forest, SVM, K Nearest Neighbour, and ANN, etc. Machine learning is emerging technology day by day in different fields.

3.1.1 Supervised learning:

In supervised learning labelled dataset is used. Here the model is trained on labelled dataset. This dataset contains both input and output parameters. Supervised learning is learning where there is input data, output data and algorithm that maps to input and output.

3.1.2. Unsupervised learning:

Unsupervised learning is learning that contains only input data and no output data is present. This is applied where there is need to model data distribution in order to get more and more data and there is no any supervisor (like teacher) to supervised the things.

3.1.3 Semi-supervised learning:

The supervised learning has disadvantage that it required labelled dataset. This process is very costly while dealing with large volume of data. Unsupervised learning also has disadvantage that the range of its application is limited.

3.1.4 Support Vector Machines:

SVM stands for Support Vector Machine. It is a simple algorithm in machine learning. It belongs to the supervised learning category in machine learning which is used for both regression and classification analysis but it is a discriminant classifier which is widely used as classification algorithm.

CHAPTER 4: PROPOSED SYSTEM

This chapter describe about process to get the result of the classification of leaf disease. Figure 3 shows the flow chart of the proposed approach. First process is getting the image acquisition by using 8-Mega Pixel smart phone. The fifty samples image for healthy leaf image and fifty sample unhealthy leaf images are taken and the image processing method is used. The process for image processing, has three components which are contrast enhancement, segmentation and features extraction. Lastly, the collected data will be classified to health or unhealthy of leaves using Artificial Neural Network.

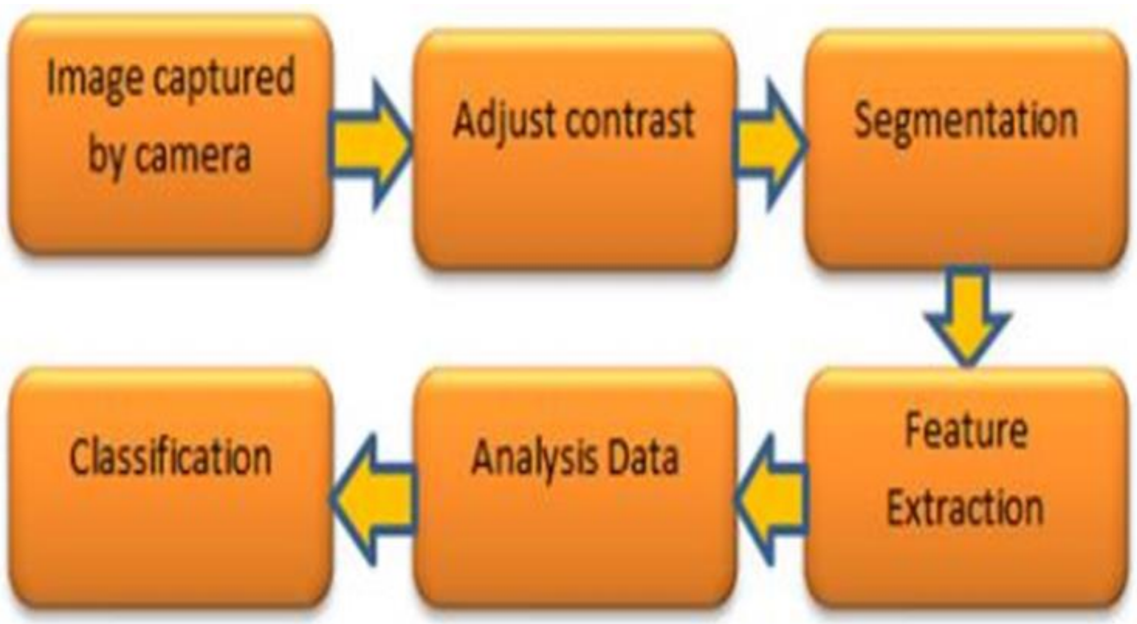


Figure 4.1 The flow chart of the proposed system

CHAPTER 5:IMPLEMENTATION

5.1 Explanation of Key Elements

5.1.1 PYTHON

Guido Van Rossum designed Python, an interpreted, high-level, general-purpose programming language that was initially released in 1991.

The Python language has the following features:

- Simple to Understand and Apply Python is a simple language to learn and use. It's a high-level programming language that's helpful to developers.
- Expressive Language Python is more expressive than other languages, making it more intelligible and readable.
- Language Interpretation Python is an interpreted language, which means that the interpreter runs the code line by line.

5.2 PYTHON LIBRARIES

NUMPY:

NumPy is a Python library that includes support for enormous, multi-dimensional arrays and matrices, as well as a large set of high-level mathematical functions that may be applied to those arrays. Numeric, the forerunner to NumPy, was designed by Jim Hugunin with help from a number of other people. Travis Oliphant built NumPy in 2005 by heavily modifying Numeric and combining features from the competitor Num array. NumPy is a large open-source software project with numerous contributors.

PANDAS:

Pandas is an open-source, BSD-certified library for the Python programming language that provides high-overall performance, easy-to-use statistics systems, and data analysis tools. Its goal is to become the most important high-level building element for undertaking realistic; real-world international records evaluations in Python panda is well-suited to a wide range of statistical applications:

- Tabular statistics containing columns of varying types, such as those seen in an SQL database or an Excel spreadsheet.
- Time collecting data that are sorted and unordered (but not necessarily at the same frequency).
- Row and column labels for arbitrary matrix information (homogeneously typed or heterogeneous)

MAT PLOT LIB:

Matplotlib is a Python 2D plotting toolkit that generates book-quality figures in a variety of hardcopy codecs and interactive contexts. Matplotlib is a Python library that may be used in scripts, the Python and I Python shells, the Jupiter Notebook, web applications servers, and four graphical user interface toolkits. Matplotlib aims to make both smooth and difficult tasks feasible.

SEABORN:

Seaborn is a data visualization package for Python that is mostly based on Matplotlib. It provides a high-level interface for creating visually beautiful and useful statistics graphs. Seaborn is a Python module for creating statistical visuals. It's based on Matplotlib, and it's tightly integrated with Panda's data systems.

Scikit-learn:

To put it another way, the sci-kit study is a free software system studying library for the Python computer language. It includes support vector machines, random forests, gradient boosting, k-method, and DBSCAN, among other categorization, regression, and clustering algorithms, and is designed to work with the Python numerical and clinical libraries NumPy and SciPy. Scikit-research was created in 2007 as a Google Summers of Code initiative by David Cornopean.

CHAPTER 6 : SOFTWARE REQUIREMENTS

HARDWARE REQUIREMENT

- System : Pentium IV 2.4 GHz.
- Hard Disk : 40GB.
- Floppy Drive : 1.44 Mb.
- Monitor : 15 VGA Colour.
- RAM : 512 Mb.

SOFTWARE REQUIREMENT

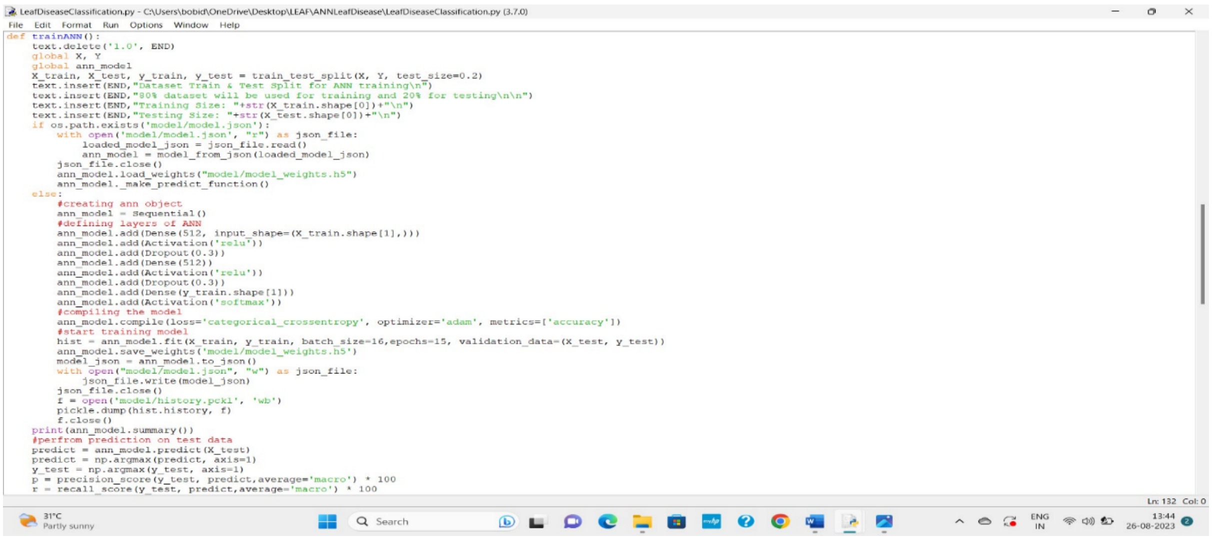
- Operating system : Windows XP/7
- Coding Language : Python
- Algorithm : Artificial Neural Network
- Tools : NumPy, Pandas, MatPlotLib, TensorFlow, SciKit-Learn, Keras, Seaborn

CHAPTER 7: RESULT AND DISCUSSIONS

In this project we are using the ANN algorithm to detect and classify leaf diseases and to train this algorithm we have used the 'Plant Village' dataset which contains 25 different types of leaf diseases including health leaf.

To implement this project we have designed the following modules

1) Upload Leaf Disease Dataset: Using this module we will upload the leaf dataset to the application.
2) Image Preprocessing: Using this module we will read all images and then resize images to equal size and then normalize all pixel values.
3) Segmentation & Features Extraction: Using this module we will apply the KMEANS algorithm to segment the infected part of the leaf and then apply the PCA features extraction algorithm to extract important features from all images.
4) Train ANN Algorithm: Extracted features will be input to the ANN algorithm to train a model and this model can be applied to test images to detect and classify diseases.
5) Disease Classification: Using this module we will upload test images and then apply the ANN model to detect disease from that image.
6) In the fig 6.1 showing code for KMEANS segmentation.



```
def trainANN():
    text.delete('1.0', END)
    global X, Y
    global ann_model
    X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.2)
    text.insert(END,"Dataset Train & Test Split for ANN training\n")
    text.insert(END,"80% dataset will be used for training and 20% for testing\n\n")
    text.insert(END,"Training Size: "+str(X_train.shape[0])+"\n")
    text.insert(END,"Testing Size: "+str(X_test.shape[0])+"\n")
    if os.path.exists('model/model.json'):
        with open('model/model.json', "r") as json_file:
            loaded_model_json = json_file.read()
            ann_model = model_from_json(loaded_model_json)
        json_file.close()
        ann_model.load_weights("model/model_weights.h5")
        ann_model._make_predict_function()
    else:
        #creating ann object
        ann_model = Sequential()
        #defining layers of ANN
        ann_model.add(Dense(512, input_shape=(X_train.shape[1],)))
        ann_model.add(Activation('relu'))
        ann_model.add(Dropout(0.3))
        ann_model.add(Dense(512))
        ann_model.add(Activation('relu'))
        ann_model.add(Dropout(0.3))
        ann_model.add(Dense(y_train.shape[1]))
        ann_model.add(Activation('softmax'))
        #compiling the model
        ann_model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
        #start training model
        hist = ann_model.fit(X_train, y_train, batch_size=16,epochs=15, validation_data=(X_test, y_test))
        ann_model.save_weights('model/model_weights.h5')
        model_json = ann_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()
    print(ann_model.summary())
    #perfrom prediction on test data
    predict = ann_model.predict(X_test)
    predict = np.argmax(predict, axis=1)
    y_test = np.argmax(y_test, axis=1)
    p = precision_score(y_test, predict,average='macro') * 100
    r = recall_score(y_test, predict,average='macro') * 100
```

Fig :6.1. K Means segmentation

In the fig 6.1 read red color comments to know about segmentation and in the fig 6.2 we have created ANN Model.

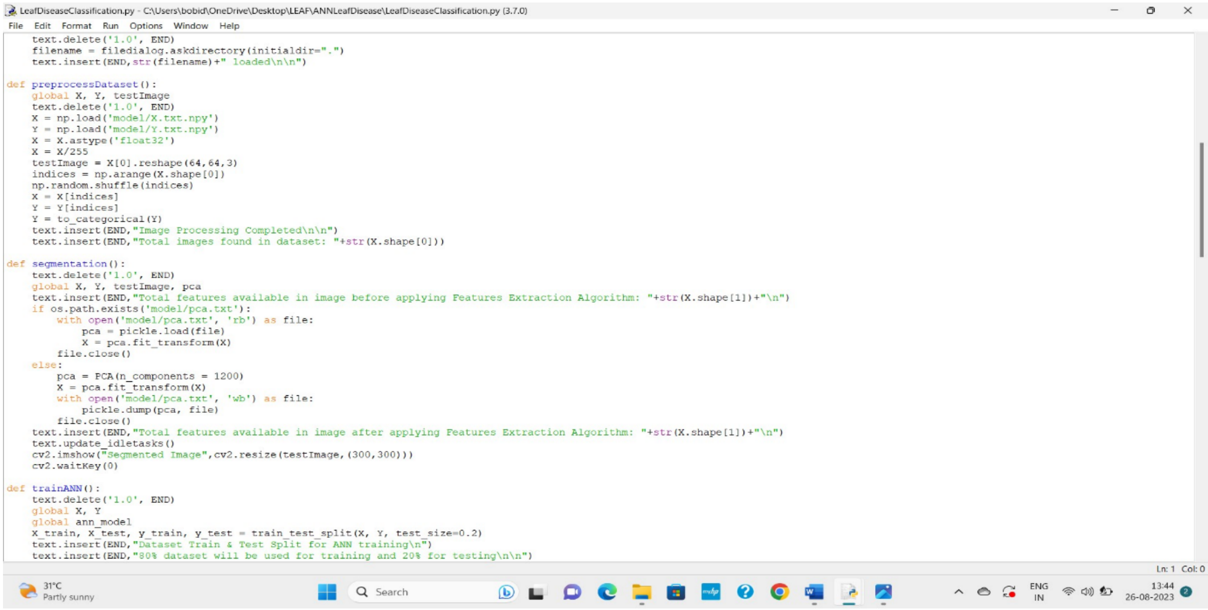


Fig:6.2. ANN Model

In the fig 6.2 read red color comments to know about ANN model. To run project double, click on 'run.bat' file to get below output.

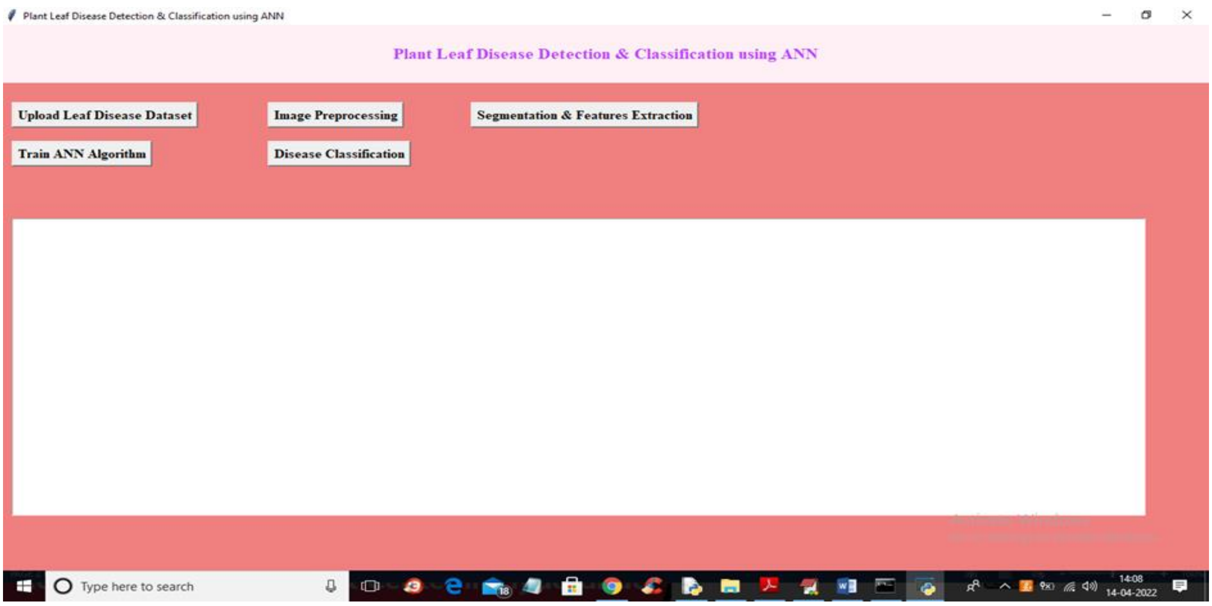


Fig:6.3. Upload leaf disease data set

In the fig 6.3 click on "upload Leaf Disease Data set" button to upload dataset and get below output

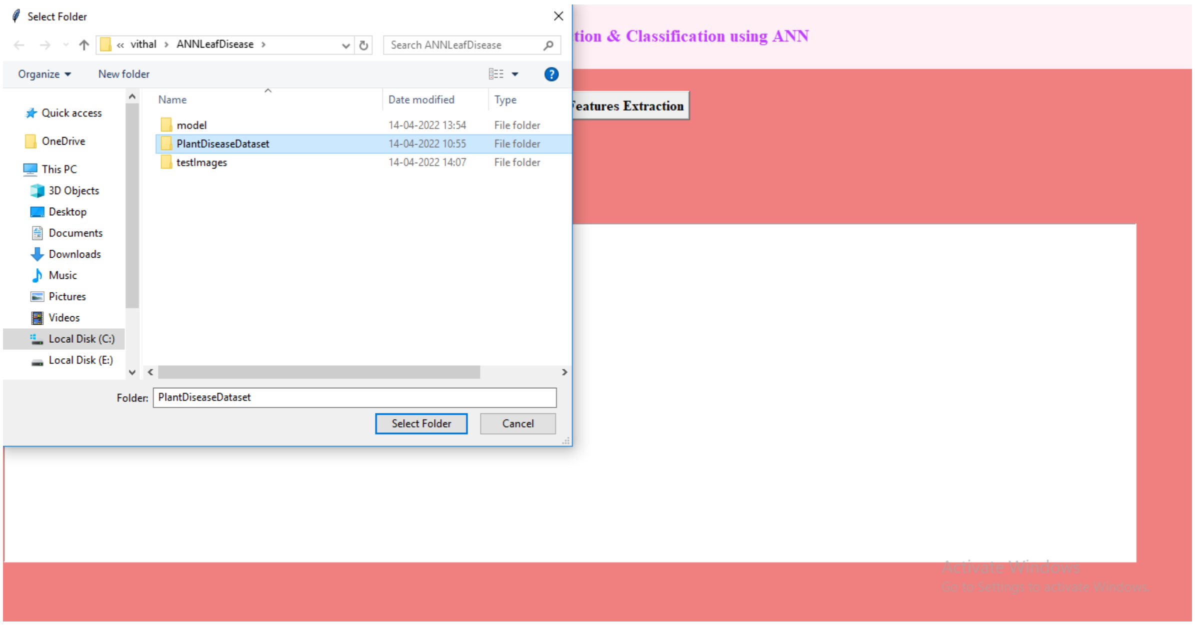


Fig:6.4. Uploading plant disease

In the fig 6.4 selecting and uploading 'Plant Disease' dataset folder and then click on 'Select Folder' button to upload dataset and to get below output.

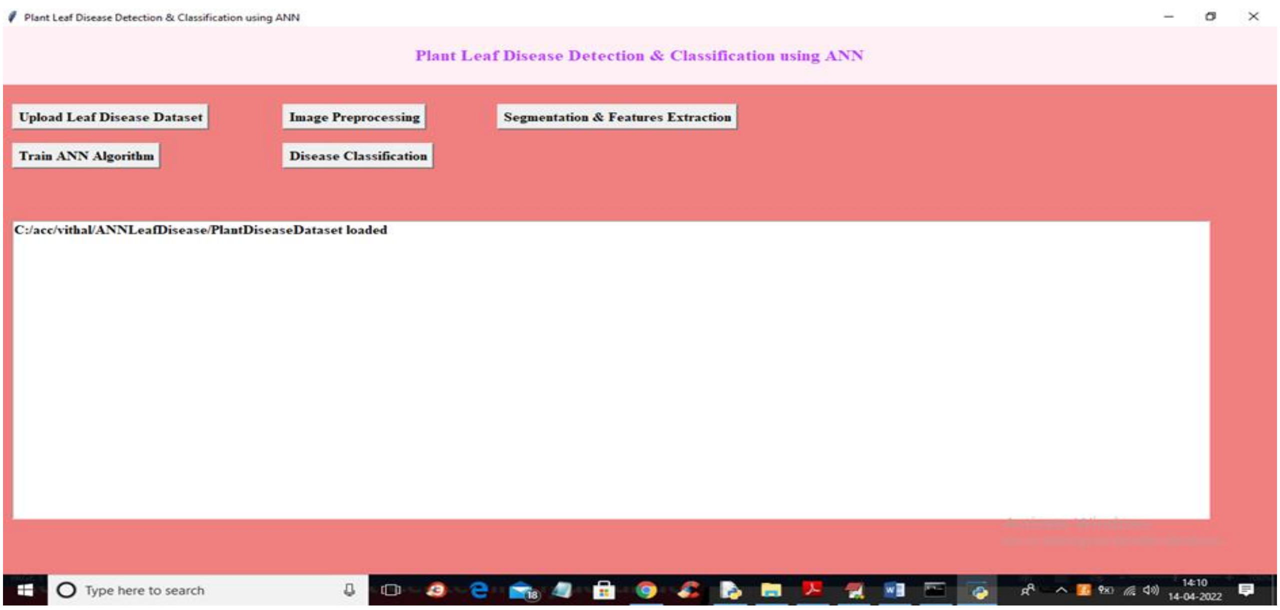


Fig:6.5. Choosing Image Processing

Now click on 'Image Processing' button to image get below output.

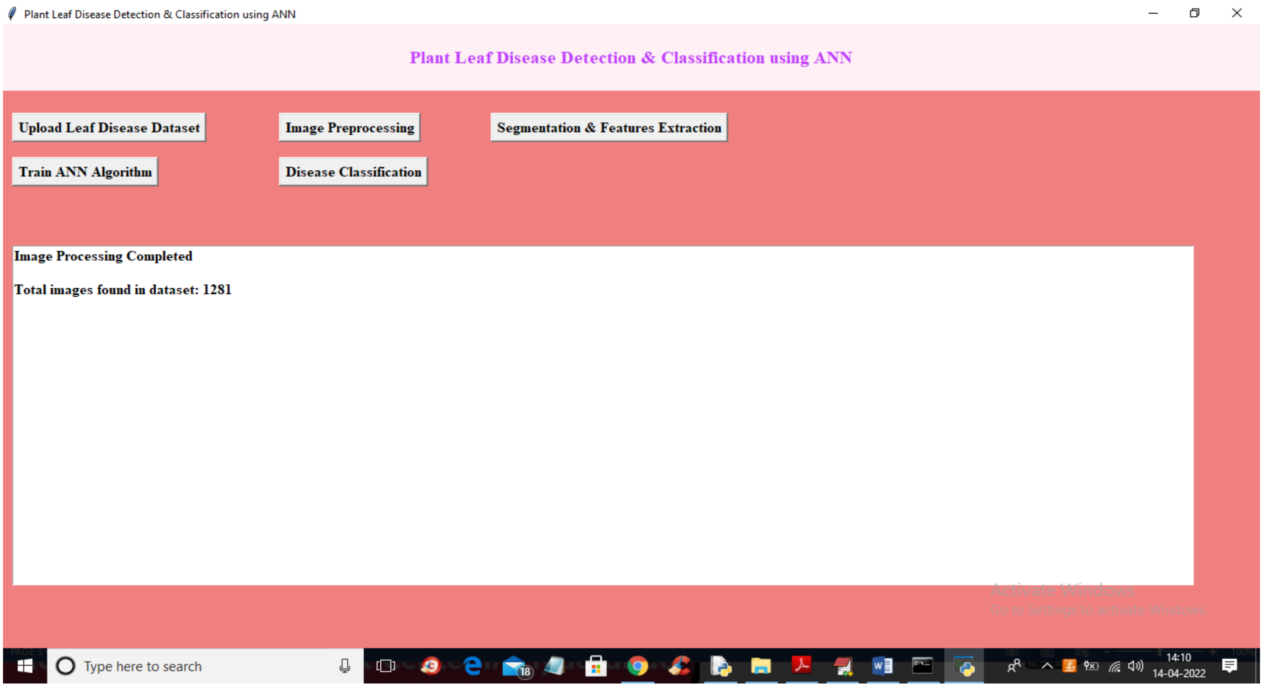


Fig:6.6. Segmentation &Features Extraction

In the fig 6.6 all 1281 images from data set are normalized and now click on 'Segmentation & Features Extraction' to segment images and then extract features and then will get below output.

Fig:6.7. Infected Part Image

In fig 6.7, we are displaying a sample segmented image where the green part was removed and taking only the infected part and now closed above image to get the below output.

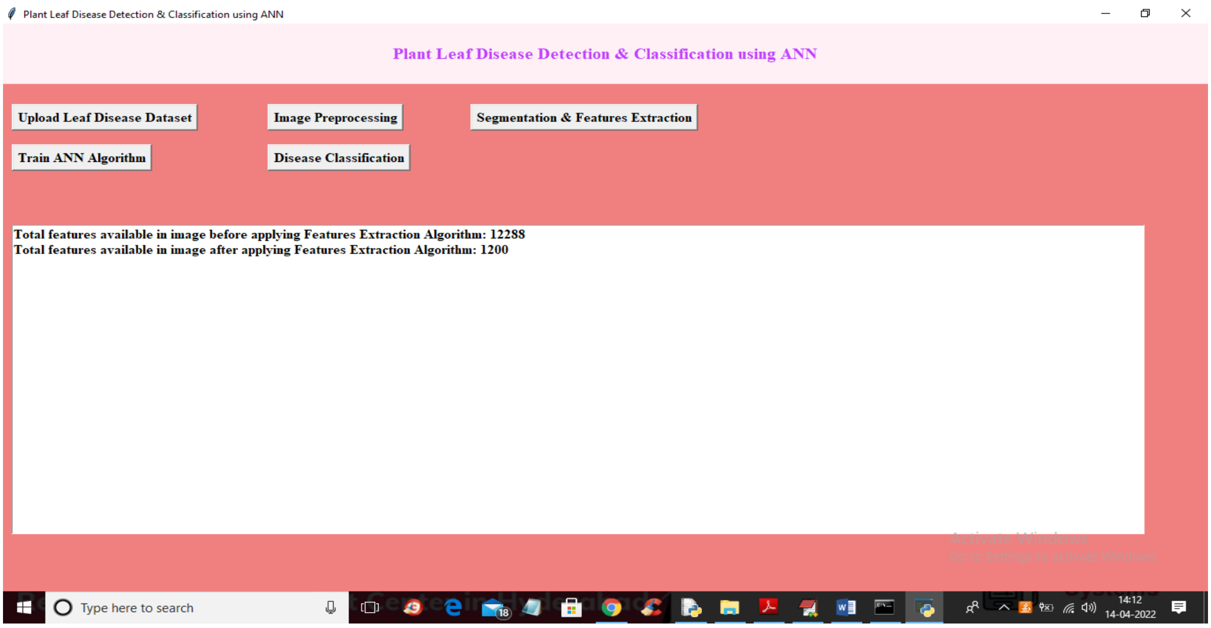


Fig:6.8 Choosing ANN Algorithm

In the fig 6.8 each image contains 12288 features and after PCA we got 1200 extracted features and now click on Train ANN Algorithm' button to train ANN with images and get below output.

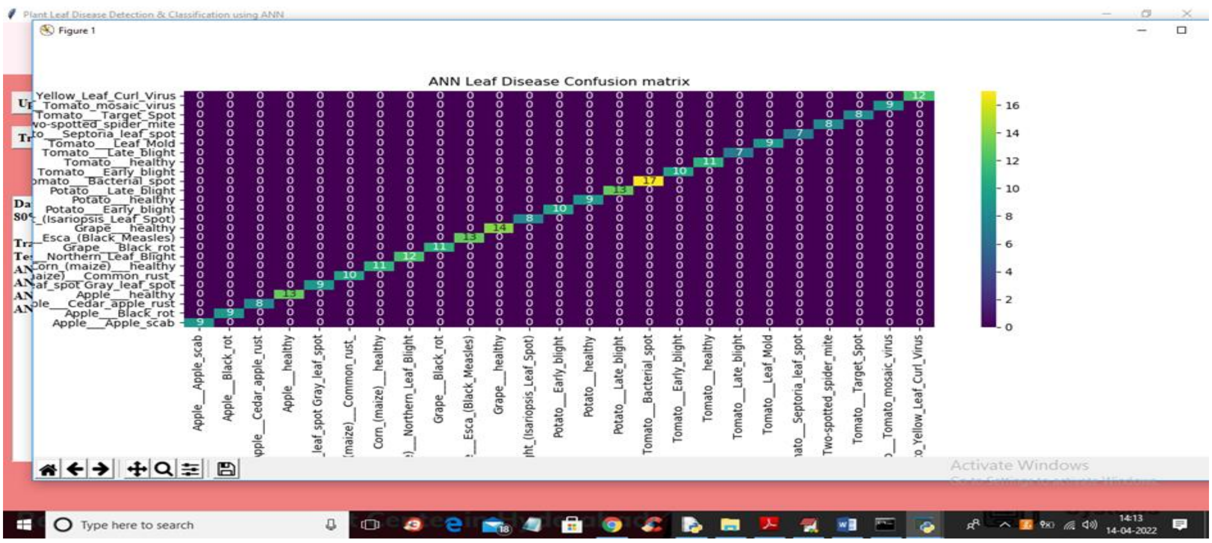


Fig:6.9. Confusion Matrix

After training, will get above confusion matrix graph on test data and in above graph x-axis represent Predicted Classes and y-axis represents TRUE classes and we can predict is accurate as it matching with the True class box and now close above graph to get below output

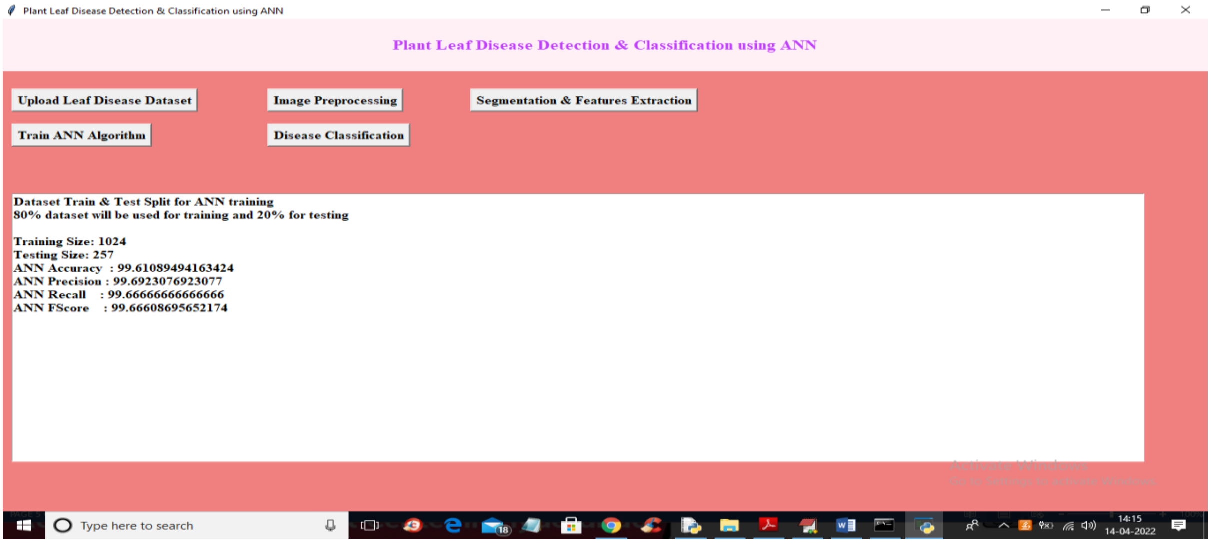


Fig:6.10 ANN Features

In the fig 6.10 displaying total images used for training and how many images are used for testing and we got 99% accuracy with ANN and now click on 'Disease Classification' button to upload test images like fig 6.11

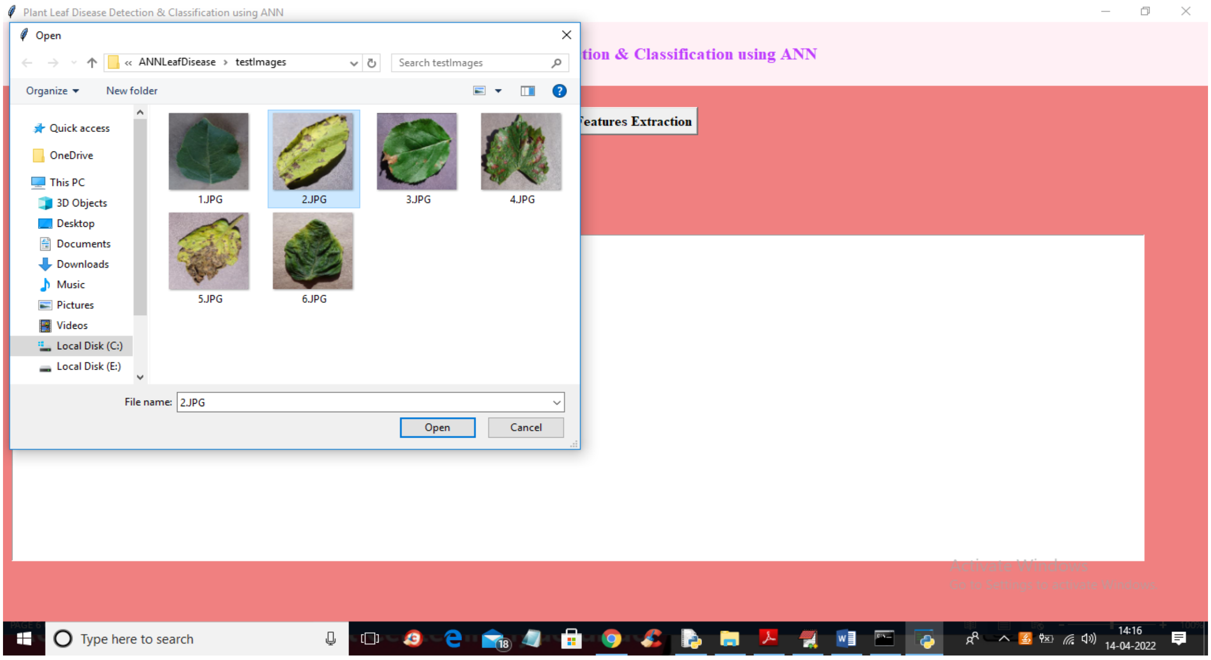


Fig:6.11. Disease Classification

In Fig 6.11 select and upload the 2.JPG file and then Click on the 'Open' button to get the below output.

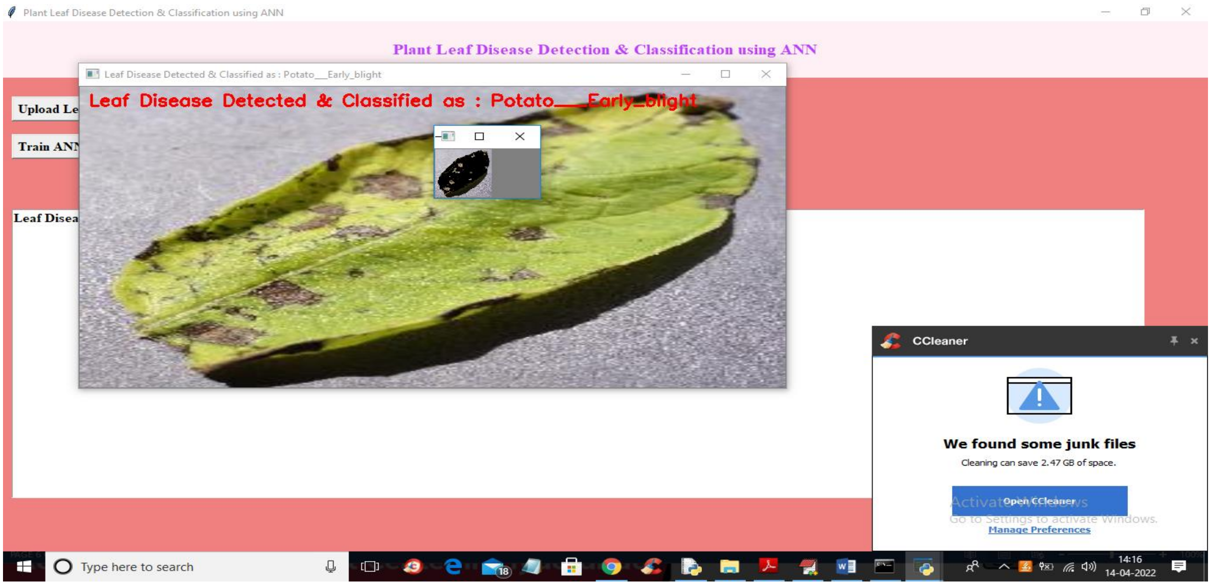


Fig:6.12. Potato early blight disease

In Fig 6.12 shows the leaf disease predicted as Potato early blight" and Tomato Septoria leaf spot.

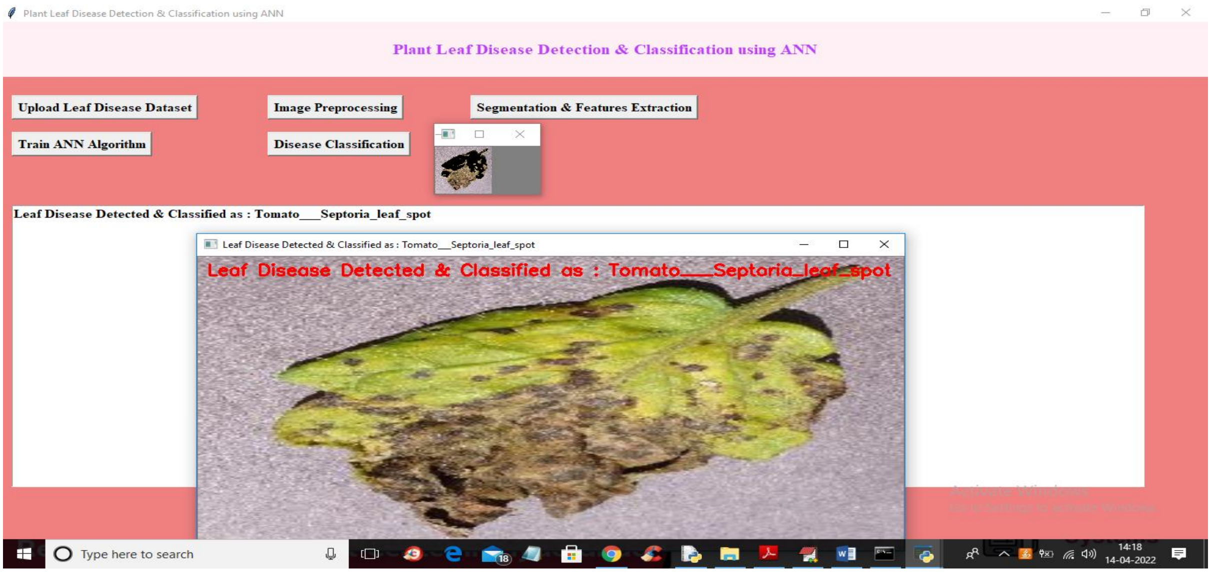


Fig:6.13 Tomato Septoria leaf spot

In Fig 6.13 shows the leaf disease predicted as Potato early blight and Tomato Septoria leaf spot.

CHAPTER 8: CONCLUSION & FUTURE SCOPE

CONCLUSION

Leaf disease classification using Artificial Neural Network has been successfully analyzed using image processing method and classified using Neural Network to get the performance of the data. The image processing method has been applied to 100 samples of leaf and the data based on color and area of unhealthy analyzed. The objective to capture and analysis data from leaf images for classify healthy or unhealthy of the leaves of medicine plants was achieved using image processing method.

In the future this methodology can be integrated with other yet to be developed methods for disease identification and classification. The use of other algorithms can be explored to enhance the efficiency of the system in future.

FUTURE SCOPE

1. **Development of Portable Devices:** The development of portable and handheld devices for disease detection is increasing at a rapid pace. In the future, there could be an increase in the production of such devices that could utilize ANN classifiers to detect and classify plant leaf diseases quickly and accurately.
2. **Integration with Smart Farming:** Smart farming is an emerging field that combines IoT devices and machine learning algorithms to monitor and optimize crop yield. The plant leaf disease detection system using ANN classifiers can be integrated with smart farming technologies to improve crop health and prevent yield loss.
3. **Expansion to Different Crops and Diseases:** Currently, the plant leaf disease detection and classification system using ANN classifier is limited to specific crops and diseases. In the future, the system can be expanded to detect and classify diseases in other crops, including fruits and vegetables.

CHAPTER 9 : APPENDIX

SOURCE CODE:

```
from tkinter import *

import tkinter

from tkinter import filedialog

import numpy as np

from tkinter.filedialog import askopenfilename

from tkinter import simpledialog

import matplotlib.pyplot as plt

import os

import numpy as np

import pandas as pd

from sklearn.model_selection import train_test_split

from sklearn.metrics import accuracy_score

from sklearn.metrics import precision_score

from sklearn.metrics import recall_score

from sklearn.metrics import f1_score

import seaborn as sns

from sklearn.metrics import confusion_matrix

import cv2

from keras.utils.np_utils import to_categorical

from keras.models import Sequential

from keras.layers import Dense, Dropout, Activation

from keras.models import model_from_json

import pickle

from sklearn.model_selection import train_test_split
```

```
from sklearn.decomposition import PCA

main = tkinter.Tk()

main.title("Plant Leaf Disease Detection & Classification using ANN")

main.geometry("1000x650")

global ann_model

global filename

global X, Y

global X_train, X_test, y_train, y_test, testImage, pca

low_green = np.array([25, 52, 72])

high_green = np.array([102, 255, 255])

leaf_labels=['Apple___Apple_scab:drug1','Apple___Black_rot',
'Apple___Cedar_apple_rust','Apple___healthy','Corn_(maize)___Cercospora
_leaf_spot'Gray_leaf_spot','Corn_(maize)___Common_rust_','Corn_(maize)_
__healthy','Corn_(maize)___Northern_Leaf_Blight','Grape___Black_rot',
'Grape___Esca_(Black_Measles)','Grape___healthy','Grape___Leaf_blight_(
Isariopsis_Leaf_Spot)','Potato___Early_blight','Potato___healthy','Pota
to___Late_blight','Tomato___Bacterial_spot','Tomato___Early_blight','To
mato___healthy',
'Tomato___Late_blight','Tomato___Leaf_Mold','Tomato___Septoria_leaf_spo
t','Tomato___Spider_mitesTwo_spotted_spider_mite','Tomato___Target_Spot
','Tomato___Tomato_mosaic_virus','Tomato___Tomato_Yellow_Leaf_Curl_Viru
s']

def loadDataset():

    global filename, dataset

    text.delete('1.0', END)

    filename = filedialog.askdirectory(initialdir=".")

    text.insert(END,str(filename)+" loaded\n\n")

    def preprocessDataset():

    global X, Y, testImage

    text.delete('1.0', END)

    X = np.load('model/X.txt.npy')

    Y = np.load('model/Y.txt.npy')
```

```
    X = X.astype('float32')

    X = X/255

    testImage = X[0].reshape(64,64,3)

    indices = np.arange(X.shape[0])

    np.random.shuffle(indices

    X = X[indices]

    Y = Y[indices]

    Y = to_categorical(Y)

    text.insert(END,"Image Processing Completed\n\n")

    text.insert(END,"Total images found in dataset: "+str(X.shape[0]))

def segmentation():

    text.delete('1.0', END)

    global X, Y, testImage, pca

    text.insert(END,"Total features available in image before applying
Features Extraction Algorithm: "+str(X.shape[1])+"\n")

    if os.path.exists('model/pca.txt'):

        with open('model/pca.txt', 'rb') as file:

            pca = pickle.load(file)

            X = pca.fit_transform(X)

        file.close()

    else:

        pca = PCA(n_components = 1200)

        X = pca.fit_transform(X)

        with open('model/pca.txt', 'wb') as file:

            pickle.dump(pca, file)

        file.close()

    text.insert(END,"Total features available in image after applying
Features Extraction Algorithm: "+str(X.shape[1])+"\n")
```

```
    text.update_idletasks()
    cv2.imshow("Segmented Image",cv2.resize(testImage,(300,300)))
    cv2.waitKey(0)
def trainANN():
    text.delete('1.0', END)
    global X, Y
    global ann_model
    X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.2)
    text.insert(END,"Dataset Train & Test Split for ANN training\n")
    text.insert(END,"80% dataset will be used for training and 20% for testing\n\n")
    text.insert(END,"Training Size: "+str(X_train.shape[0])+"\n")
    text.insert(END,"Testing Size: "+str(X_test.shape[0])+"\n")
    if os.path.exists('model/model.json'):
        with open('model/model.json', "r") as json_file:
            loaded_model_json = json_file.read()
            ann_model = model_from_json(loaded_model_json)
        json_file.close()
        ann_model.load_weights("model/model_weights.h5")
        ann_model._make_predict_function()
    else:
        ann_model = Sequential()
        ann_model.add(Dense(512, input_shape=(X_train.shape[1],)))
        ann_model.add(Activation('relu'))
        ann_model.add(Dropout(0.3))
        ann_model.add(Dense(512))
        ann_model.add(Activation('relu'))
```

```
        ann_model.add(Dropout(0.3))

        ann_model.add(Dense(y_train.shape[1]))

        ann_model.add(Activation('softmax'))

        ann_model.compile(loss='categorical_crossentropy',
optimizer='adam', metrics=['accuracy'])

        hist = ann_model.fit(X_train, y_train, batch_size=16,epochs=15,
validation_data=(X_test, y_test))

        ann_model.save_weights('model/model_weights.h5')

        model_json = ann_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()

    print(ann_model.summary()

    predict = ann_model.predict(X_test)

    predict = np.argmax(predict, axis=1)

    y_test = np.argmax(y_test, axis=1)

    p = precision_score(y_test, predict,average='macro') * 100

    r = recall_score(y_test, predict,average='macro') * 100

    f = f1_score(y_test, predict,average='macro') * 100

    a = accuracy_score(y_test,predict)*100

    text.insert(END,'ANN Accuracy  : '+str(a)+"\n")

    text.insert(END,'ANN Precision : '+str(p)+"\n")

    text.insert(END,'ANN Recall    : '+str(r)+"\n")

    text.insert(END,'ANN FScore    : '+str(f)+"\n\n")

    text.update_idletasks()
```

```
    LABELS = leaf_labels

    conf_matrix =confusion_matrix(y_test, predict) #calculate confusion
matrix

    plt.figure(figsize =(16, 6))

    ax =sns.heatmap(conf_matrix, xticklabels = LABELS, yticklabels =
LABELS, annot = True, cmap="viridis" ,fmt ="g");

    ax.set_ylim([0,len(LABELS)])

    plt.title("ANN Leaf Disease Confusion matrix")

    plt.ylabel('True class')

    plt.xlabel('Predicted class')

    plt.show()

    def classification():

      global ann_model, pca

      text.delete('1.0', END)

      filename = filedialog.askopenfilename(initialdir="testImages")

      img = cv2.imread(filename) #read input image

       Z = np.float32(img.reshape((-1,3))) #create Z value from image

    criteria=(cv2.TERM_CRITERIA_EPS+cv2.TERM_CRITERIA_MAX_ITER,10,1.0)

 K=4
_,labels,centers=cv2.kmeans(Z,K,None,criteria,10,cv2.KMEANS_RANDOM_CENT
ERS)

    labels = labels.reshape((img.shape[:-1]))

    reduced = np.uint8(centers)[labels]

    img=cv2.resize(img, (64,64), interpolation=cv2.INTER_CUBIC)

    imgHSV = cv2.cvtColor(img, cv2.COLOR_BGR2HSV)

    mask = cv2.inRange(imgHSV, low_green, high_green)

    mask = 255-mask #masking done here

    res = cv2.bitwise_and(img, img, mask=mask)

    segmented = res #get segmented image
```

```
    res = res.ravel()

    test = []

    test.append(res)

    test = np.asarray(test)

    test = pca.transform(test) #extract features using PCA

    print(test.shape)

    test = test.astype('float32') #normalized the pixel values

    test = test/255

    preds = ann_model.predict(test)#predict the disease using ann model

    predict = np.argmax(preds)

    print(predict)

    img = cv2.imread(filename)

    img = cv2.resize(img, (800,400))

    text.insert(END,'Leaf   Disease   Detected   &   Classified   as   :
'+leaf_labels[predict]+"\n")

    text.update_idletasks()

    cv2.putText(img,   'Leaf   Disease   Detected   &   Classified   as:
'+leaf_labels[predict],(10,25),cv2.FONT_HERSHEY_SIMPLEX,0.7,(0,0,255),2
)

    cv2.imshow('Leaf    Disease    Detected    &    Classified    as    :
'+leaf_labels[predict], img)

    cv2.imshow("Segmented Image",cv2.resize(segmented,(200,200)))

    cv2.waitKey(0)

font = ('times', 15, 'bold')

title = Label(main,text='Plant Leaf Disease Detection & Classification
using ANN', justify=LEFT)

title.config(bg='lavender blush', fg='DarkOrchid1')

title.config(font=font)

title.config(height=3, width=120)
```

```
title.place(x=100,y=5)

title.pack()

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

loadButton = Button(main, text="Upload Leaf Disease Dataset", command=loadDataset)

loadButton.place(x=10,y=100)

loadButton.config(font=font1)

preprocessButton=Button(main,text="Image Preprocessing", command=preprocessDataset)

preprocessButton.place(x=300,y=100)

preprocessButton.config(font=font1)

segButton = Button(main, text="Segmentation & Features Extraction", command=segmentation)

segButton.place(x=530,y=100)

segButton.config(font=font1)

annButton = Button(main, text="Train ANN Algorithm", command=trainANN)

annButton.place(x=10,y=150)

annButton.config(font=font1)

clsButton=Button(main,text="DiseaseClassification",command=classificati
on)

clsButton.place(x=300,y=150)

clsButton.config(font=font1)

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

text=Text(main,height=20,width=160)

scroll=Scrollbar(text)

text.configure(yscrollcommand=scroll.set)

text.place(x=10,y=250)

text.config(font=font1) main.config(bg='light coral')

main.mainloop()
```

CHAPTER 10: BIBLIOGRAPHY

1. V. Rubesh, S. S. Bharathi, G. Santhiya, and R. Anuradha, "A Comparison of Deep Learning Algorithms for Plant Disease Classification," in Advances in Cybernetics, Cognition, and Machine Learning for Communication Technologies, Lecture Notes in Electrical Engineering, vol. 643, pp. 153- 161, Springer, Singapore, 2020.
2. Savary, Serge, et al. The global burden of pathogens and pests on major food crops." Nature ecology & evolution 3.3 (2019): 430.
3. A. Ramcharan, P. McCloskey, K. Baranowski et al., "A mobile-based deep learning model for cassava disease diagnosis," Frontiers in Plant Science, vol. 10, p. 272, 2019.
4. K. Nagpal, D. Foote, Y. Liu et al., Publisher correction: development and validation of a deep learning algorithm for improving Gleason scoring of prostate cancer," NPJ Digital Medicine, vol. 2, no. 1, p. 113, 2019.
5. S. Verma, A. Chug, A. P. Singh, S. Sharma, and P. Rajvanshi, "Deep Learning-Based Mobile Application for Plant Disease Diagnosis: A Proof of Concept with a Case Study on Tomato Plant," in Applications of Image Processing and Soft Computing Systems in Agriculture, pp. 242-271, IGI Global, 2019.
6. Fujita, E., et al. "A practical plant diagnosis system for field leaf images and feature visualization. International Journal of Engineering & Technology 7.4.11 (2018): 49-54.
7. Kaminari and F. X. Prenafeta-Boldú, "Deep learning in agriculture: a survey." Computers and Electronics in Agriculture, vol. 147, pp. 70-90,2018.
8. 8.B. K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis, Computers and Electronics in Agriculture, vol. 145, pp. 311-318, 2018.
9. J. G. A. Braedon, "Factors influencing the use of deep learning for plant disclose recognition," Biosystems Engineering, vol. 172, pp. 84-91, 2018.

10 . Dhakal and S. Shakya, "Image-based plant disease detection with deep learning," International Journal of Computer Trends and Technology, vol. 61, no. 1, pp. 26-29, 2018.

11 . S. Wallelign, M. Polceanu, and C. Buche, "Soybean plant disease identification using convolutional neural network," in Proc. Thirty-First International Florida Artificial Intelligence Research Society Conference (FLAIRS-31), pp. 146-151, Melbourne, FL, USA, 2018,

12 . T. Wiesner-Hanks, E. L. Stewart, N. Kaczmar, "Image set for deep learning: field images of maize annotated with disease symptoms," BMC Research Notes, vol. 11, no. 1, p. 440, 2018.

13 . G. Wang, Y. Sun, and J. Wang, "Automatic image-based plant disease severity estimation using deep learning." Computational Intelligence and Neuroscience, vol. 2017, Article ID 2917536, 8 pages, 2017.

14 . Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes, Deep learning for image-based cassava disease detection," Frontiers in Plant Science, vol. 8, article 1852, 2017.

15 . Mohanty, Sharada P., David P. Hughes, and Marcel Salathé. "Using deep learning for image-based plant disease detection." Frontiers in plant science 7 (2016): 1419.

16 . Szegedy, Christian, et al. "Going deeper with convolutions." Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

17 . Deng, Jia, et al. "Image net: A large-scale hierarchical image database." 2009 IEEE conference on computer vision and pattern recognition. 2009.

18 . Cunningham, Padraig, and Sarah Jane Delany. "k-Nearest neighbor classifiers. Multiple Classifier Systems 34.8 (2007): 1-17.

19 . Duan, Kai-Bo, and S. Sathya Keerthi. "Which is the best multiclass SVM method? An empirical study." International workshop on multiple systems. Springer, Berlin, Heidelberg, 2005

20 . Cortes, Corinna, and Vladimir Vapnik. “Support-vector networks.” Machine learning 20.5(1995): 273-297