A

Major Project

On

DETECTION AND CLASSIFICATION OF FRUIT DISEASES USING IMAGE PROCESSING AND CLOUD COMPUTING

(Submitted in partial fulfillment of the requirements for the award of Degree)

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE AND ENGINEERING

By

Chevallamudi Mohan Krishna	(167R1A0523)
H.Ajay Reddy	(177R1A0527)
Zaladanki Lakshmi Sunarvitha	(177R1A0556)
Bussa Nikhitha	(177R1A05C7)

Under the Guidance of

DR.A.PRABHU

(Associate Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi) Recognized Under Section 2(f) & 12(B) of the UGCAct.1956,

Kandlakoya (V), Medchal Road, Hyderabad-501401

2018-2022

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR
GROUP OF INSTITUTIONS
EXPLORE TO INVENT

CERTIFICATE

This is to certify that the project entitled "**DETECTION AND CLASSIFICATION OF FRUIT DISEASES USING IMAGE PROCESSING AND CLOUD COMPUTING**" being submitted by **CHEVALLAMUDI MOHAN KRISHNA(167R1A0523), H.AJAY REDDY(177R1A0527), ZALADANKI LAKSHMI SUNARVITHA (177R1A0556) & BUSSA NIKHITHA (177R1A05C7)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by him/her under our guidance and supervision during the year 2021-2022

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

Dr.A.Prabhu
Associate Professor
INTERNAL GUIDE

Dr.A.Raji Reddy
DIRECTOR

Dr. K. Srujan Raju
HOD

EXTERNAL EXAMINER

Submitted for viva voice Examination held on ____________

ACKNOWLEGDEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide.**Dr.A.Prabhu,** Associate Professor for his exemplary guidance, monitoring and constant encouragement throughout the project work. The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark. We also take this opportunity to express a deep sense of gratitude to Project Review Committee(PRC) coordinators **Mr.A.Udaykiran, Mr.J.Narsimharao, Dr.T.S.Mastan Rao , Mrs.G.Latha** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages

We are also thankful to **Dr. K. Srujan Raju**, Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy,** Director for being cooperative throughout the course of this project.We also express our sincere gratitude to **Sri. Ch. Gopal Reddy,** Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

CHEVALLAMUDI MOHAN KRISHNA(167R1A0523)
H.AJAY REDDY(177R1A0527)
ZALADANKI LAKSHMI SUNARVITHA(177R1A0556)
BUSSA NIKHITHA(177R1A05C7)

ABSTRACT

Fruit disease detection is vital at early stage since it will affect the agricultural field. In this project, mainly consider the detection and analysis of fruit infections which is available in the plant areas and storage of data about the agricultural filed and details of farmers in database and recovering the data using Cloud computing. There are more fruit diseases which occur due to the surrounding conditions, mineral levels, insects in the farm area and other factors. The detected data from the plant area is determined by image processing and stored in the database. The database consists of data related to plant leaves, fruit conditions and the symptoms of disease to be affected. The fruit details and the identification of disease from the feature extraction are stored in the database. The entire database is viewed and compared with the captured image. The mobile application is developed for processing the data and providing intimation to the farmers. Thus the variation in image from the database and also indicates the disease in the fruits

LIST OF FIGURES/TABLES

LIST OF SCREEN SHOTS

TABLE OF CONTENTS

1.INTRODUCTION

1.1 PROJECT SCOPE:

Agriculture has been the base for every people. It is most important that more than 70% of the people depend on agriculture for their livelihood in India. Nowadays the growth of productivity of plants, crops and fruits are normally affected by the diseases. The disease is a major problem arising in an agricultural field. In plants, most of the leaves and fruits are affected by diseases due to bacteria and virus. This technique is used to determine the infection on leaves, fruits and stem of the plants. In order to generate an automated database to examine the infections using proposed method.

1.2 PROJECT FEATRURES:

The various features of few fruits were initially extracted and segment the respective images. After comparison with feature values, the various disease names are analyzed and the optimal disease for the image is identified and the disease is indicated by an alert box and can be provided as the message through mobile application..

2. LITERATURE SURVEY

Pratik Agarwal, “AGROCLOUD-Open surveillance of Indian Agriculture via cloud”2016.International Conference on Information Technology(InCITe)-The Next Generation IT Summit. Athmaja S1, Hanumanthappa M2, “Applications of Mobile Cloud Computing and Big Data Analytics in Agriculture Sector- A Survey”. International Journal of Advanced Research in Computer and Communication Engineering ICRITCSA M S Ramaiah Institute of Technology, Bangalore Vol. 5, Special Issue 2, October 2016.Li Tan1,2†,Hongfei Hou1,Qin Zhang2.“An Extensible Software Platform for Cloud-based Decision Support and Automation in Precision Agriculture”.2016 IEEE 17th International Conference on Information Reuse and Integration. Li Tan1,2†,Hongfei Hou1,Qin Zhang2.“An Extensible Software Platform for Cloud-based Decision Support and Automation in Precision Agriculture”.2016 IEEE 17th International Conference on Information Reuse and Integration. Shiv Ram Dubey1, Pushkar Dixit2, Nishant Singh3, Jay Prakash Gupta4, “Infected Fruit Part Detection Using K-Means Clustering Segmentation Technique”. International Journal of Artificial Intelligence and Interactive Multimedia, Vol. 2, Nº 2. However, there has been a scarcity of resources for Indian farmers to carry out their operations, which lead to a chain of massive suicides between year 1997 to 2004, that average one farmer died in every 31 minutes and the scenario is worse in current times. Cloud Computing, or simply “cloud”, is the delivery of ondemand computer resources, from applications to data centers and servers, over the Internet on a pay-as-you-go basis. Example, SoundCloud, which provides access to world music to everyone is one-click.

The Agriculture Census of India surveys the land land employed for farming in 3 phases over 7% villages per tehsil and then it estimates the rest of the area, however, precise and extensive computation is required to take advantage out of this data. A cloud based scheme for the betterment of Japansese farmers and Japanese agriculture, has been proposed by the FUJITSU Sci. Tech. J. Another similar idea was proposed in IJERT, namely AgroCloud which used app-based surveillance, cloud’s monitoring as a service (MaaS), where farmers get their recommendations using agri-application. This scheme can also provide how much a farmer harvested and how he sold his harvest, and

the record can help avoiding hoarding cases, like the sudden price rise in onions case in 2010. It is found that even though mobile phones play a significant role in improving farm productivity and rural incomes, the following aspects-the quality of information, timeliness of information and trustworthiness of information- have great potential to meet the needs and expectations .Mobile computing technology allows transmission of data through wireless enabled devices such as mobile phones, laptops, PDA, etc. without having to be connected to a fixed physical link. By integrating cloud computing into the mobile environment, the MCC technology overcomes many obstacles such as battery life, storage and bandwidth, heterogeneity, scalability and availability and security. . Manav Singhal et al. developed a mobile application called “Krishi Ville” to help farmers. Similarly, Krishi Vigyan Kendra contact addresses are also hard coded in the application. Shitala Prasad et al. proposed an MCC model called Agro Mobile on a handheld device which helps farmers for relatively better cultivation and marketing.

The accuracy of this method is more than 90 % on test data. Mayank Agarwal et al. proposed a mobile application named “Agro-App”.We require a good understanding and knowledge about plants and identify new and rare species to increase the agricultural productivity and also to support drug industry We employ machine learning because some plants cannot be identified or are not distinguishable except for sometimes when they bloom flowers or bear fruit. Also a lot of work has been done in identifying plants mainly in computer vision by using features of leaf such as color, diameter, physiological width, physiological length, perimeter ratio, aspect ratio, rectangularity but it is not up to the mark because leaves can be very same in all these features like lemon and sweet lime or peach and plum leaves. We have prepared our own dataset with full annotation which contains the images of the following plants like arahar, brinjal, cotton , chilly, custard apple, lemon, peanut, mango, tomato, sweet lime etc. They have used color based modelling for the colored or nongreen leaves and shape based modelling simple green and compound green leaves. Application will analyze the features of leaf and identify species of the plant at realtime. They have an accuracy of 95% and they gives results within few minutes. Their proposed method can achieve more accuracy than state-of-the-art methods and 170 times faster speed than this. Hence the image is rescaled and cropped out for the central 256×256 patch from the resulting image. Multiscaling is performed as it prevents overfitting yielding better results. As a result, gray component for each pixel is computed from the color image by applying the equation shown below. The proposed system was implemented using Intel Core i7 processor at 2.20 GHZ speed and the code was written using Python. Table 1.presents confusion matrix, where TP stands for True Positive, FN stands for False Negative, FP stands for True Positive and TN stands for True Negative. Many DSSs are designed to support the concept of precision agriculture which seeks to provide a holistic approach to assist farmers with optimising resources . McBratney et al. defined PA as the “kind of agriculture that increases the number of decisions per unit area of land per unit time with associated net benefits”. In the history of agricultural systems, user-driven data and model development has played an essential role in meeting users’ analysis need for decision-support. In the history of agricultural systems, user-driven data and model development has played an essential role in meeting users’ analysis need for decision-support. This black-box nature often leads to trust issues, notably when suggestions coming from a DSS fail. Moreover, an increase in the use of farm sensors, high-tech harvesters and drones, among others, has created a massive amount of data that might be difficult for decision-makers to grasp. Users, therefore, need additional tools for understanding and interpreting their data.

Visualisation is a powerful technique to address these issues and has demonstrated its usefulness in PA to communicate uncertainty from both the data and the models. As Rind et al. suggest, visualisation tools and techniques “combine the processing power of modern computers with human cognition and visual abilities to better support analysis tasks”. In line with participatory DSS development, visualisations can be especially practical and useful when following user-centred design approaches. In fact, a number of qualitative PA studies highlighted the importance of using a participatory-design approach, where the farmer's perspective is used as the central focus of the design. A key goal of such an approach is to support complex data analysis by providing diverse visualisation techniques and adapting the tool according to the farmer needs . Previous literature reviews in agriculture have focused on similar areas such as DSSs and PA, but not on visualisation techniques. For example, Kamilaris et al. presented a survey on the recent practices of big data analysis in agriculture that would help farmers and companies to extract value from data, improving their productivity. . A different systematic review highlighted the methodologies that can be used to support farmers in designing innovative agricultural production systems. In the context of precision agriculture, Imam et al. reviewed design issues for wireless sensor networks and showed a comparison of different characteristics of humidity sensors, such as sensor type, their sensitivity and power consumption, which play a key role for wireless sensor networks for precision agriculture applications. HCI is a discipline in which users as well as technology and environment are emphasised in the design process . A recent study explored the reasons behind the lack of interest by farmers in the computer-based support systems and highlighted the importance of HCI perspectives when designing DSSs. The study also found clear benefits of designing a DSS that is easy to use, fits the existing workflow of users, performs well and incubate trust, which are some of the most commonly accessed metrics of HCI research.

The eligibility criteria were specified according to the PICO framework, stated in the preferred reporting items for systematic review and meta-analysis statement. The produced keywords are presented in Table 1. Using the defined keywords, hand searching of relevant authors and articles was conducted in Google Scholar. Clustering based image segmentation methods are also used by many researchers. Medicine, content-based image and video retrieval, industrial automation, document analysis and quality control . The efficiency of color image segmentation may significantly influence the quality of an image understanding system. A detail review on various image segmentation techniques are provided by Pal & Pal. For example, image segmentation on the basis of region merging is analogue of agglomerative clustering [6]. Graph cut methods such as normalized cut and minimal cut characterize the problem of clustering in a graph theoretic way. Soft computing techniques have been used for segmenting color image by Sowmya and Sheelarani. . Borji et al. presented CLPSO-based Fuzzy color image segmentation. Cheng et al. used Fuzzy homogeneity approach for the segmentation of color image. Besides this, Genetic algorithm and artificial neural network techniques also have been used for the image segmentation .

Automatic image segmentation by integrating seeded region growing and color edge detection was proposed by Fan et al. . Another method using seeded region growing was proposed by Adams and Bischof. Shih and Cheng proposed another image segmentation method using regions in the image where based on the standard deviation in a neighbor, initial seeds are selected. . Authors in, have used the concept of k-means clustering for background subtraction. K-means is a typical clustering algorithm. Each cluster is represented by an adaptively changing center, starting from some initial values named seed-points. K-means clustering computes the distances between the inputs and centers, and assigns inputs to the nearest center. K-means method is an unsupervised clustering method that classifies the input data objects into multiple classes on the basis of their inherent distance from each other.

3. SYSTEM ANALYSIS

3.1 PROBLEM DEFINITION:

Our problem statement deals**"Detection and classification of fruit diseases''.** It states/defines that predicting the type of fruit diseases using image processing techniques. It demonstrates how image processing technology can be used for detection of fruit diseases using python. This statement clearly explains what type of disease a fruit has.

3.2 EXISTING SYSTEM:

A K-means segmentation is used for partitioning the leaf image into four clusters using the squared Euclidean distances. The method applied for feature extraction is Colour Co-occurrence method for both colour and texture features. Finally, classification is completed using neural network detection algorithm based on back propagation methodology

3.2.1 LIMITATIONS OF EXISTING SYSTEM:

Farmers are using sensors and soil sampling to collect data and this data is stored on-farm management systems that allow for better processing & analysis

3.3 PROPOSED SYSTEM:

The classification and segmentation of fruit images were performed using K-Means Algorithm and SVM technique. The various features of few fruits were initially extracted and segment the respective images. After comparison with feature values, the various disease names are analysed and the optimal disease for the image is identified and the disease is indicated by an alert box and can be provided as the message through mobile application.

3.3.1 ADVANTAGES OF PROPOSED SYSTEM:

Agriculture is becoming digital, AI in agriculture is emerging in three major categories which are agricultural robotics, soil & crop monitoring, and predictive analytics,.

3.4 FEASIBILITY STUDY:

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. Three key considerations involved in the feasibility analysis

- Economic Feasibility
- Technical Feasibility

3.4.1 ECONOMICAL FEASIBILITY :

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it give an indication of the system is economically possible for development.

3.4.2 TECHNICAL FEASIBILITY :

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the "**DETECTION AND CLASSIFICATION OF FRUIT DISESASES"**available technical resources. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

3.5 HARDWARE AND SOFTWARE REQUIREMENTS

3.5.1 HARDWARE REQUIREMENTS:

- Processor: I3
- RAM: 4GB
- Hard Disk: 128 GB

3.5.2 SOFTWARE REQUIREMENTS:

- Operating System: Windows 10
- Server-side Script: Python 3.7
- IDE: PyCharm
- Libraries Used: Pandas, Numpy,Flask
- Google colab

4. ARCHITECTURE

4.1 PROJECT ARCHITECTURE:

The project architecture is as follows:

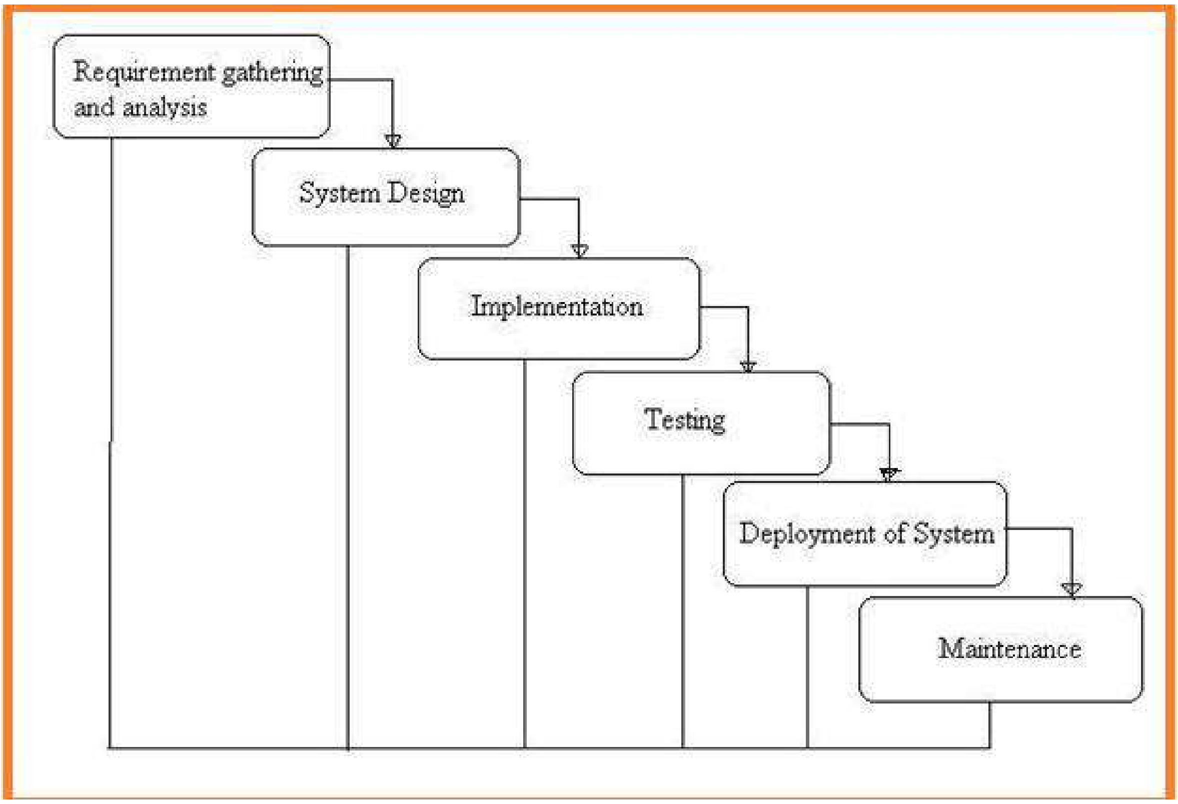


Fig 4.1: Project architecture

- Project Requisites Accumulating and Analysis
- Application System Design
- Practical Implementation
- Manual Testing of My Application
- Application Deployment of System
- Maintenance of the Project

4.1.1 SYSTEMDESIGN

In System Design has divided into three types like GUI Designing, UML Designing with avails in development of project in facile way with different actor and its utilizer case by utilizer case diagram, flow of the project utilizing sequence, Class diagram gives information about different class in the project with methods that have to be utilized in the project if comes to our project our UML Will utilizable in this way The third and post import for the project in system design is Data base design where we endeavor to design data base predicated on the number of modules in our project

4.1.2 IMPLEMENTATION

The Implementation is Phase where we endeavor to give the practical output of the work done in designing stage and most of Coding in Business logic lay coms into action in this stage its main and crucial part of the project

4.1.3 TESTING UNITTESTING

It is done by the developer itself in every stage of the project and fine-tuning the bug and module predicated additionally done by the developer only here we are going to solve all the runtime errors

4.1.4 MANUAL TESTING

As our Project is academic Leave, we can do any automatic testing so we follow manual testing by endeavor and error methods

4.1.5 DEPLOYMENT OF SYSTEM AND MAINTENANCE

Once the project is total yare, we will come to deployment of client system in genuinely world as its academic leave we did deployment i our college lab only with all need Software's withhaving Windows OS.

The Maintenance of our Project is one-time process only

4.1.6 FUNCTIONALREQUIREMENTS

1.Data Collection

2.Data Preprocessing

3.Training And Testing

4.Modiling

5.Predicting

4.1.7 NON FUNCTIONALREQUIREMENTS

NON-FUNCTIONAL REQUIREMENT (NFR) specifies the quality attribute of a software system. They judge the software system based on Responsiveness, Usability, Security, Portability and other non-functional standards that are critical to the success of the software system. Example of nonfunctional requirement, *"how fast does the website load?"* Failing to meet non-functional requirements can result in systems that fail to satisfy user needs. Non- functional Requirements allows you to impose constraints or restrictions on the design of the system across the various agile backlogs. Example, the site should load in 3 seconds when the number of simultaneous users are > 10000. Description of non-functional requirements is just as critical as a functional requirement.

- Serviceability requirement
- Manageability requirement
- Recover ability requirement
- Security requirement
- Data Integrity requirement
- Capacity requirement
- Availability requirement
- Scalability requirement
- Interoperability requirement
- Reliability requirement
- Maintainability requirement
- Regulatory requirement
- Environmental requirement

EXAMPLES OF NON-FUNCTIONAL REQUIREMENTS

Here, are some examples of non-functional requirement:

- Users must upload dataset
- The software should be portable. So moving from one OS to other OS does not create any problem.
- Privacy of information, the export of restricted technologies, intellectual property rights, etc. should be audited.

ADVANTAGES OF NON-FUNCTIONAL REQUIREMENT

Benefits/pros of Non-functional testing are:

- The nonfunctional requirements ensure the software system follow legal andcompliance rules.
- They ensure the reliability, availability, and performance of the software system
- They ensure good user experience and ease of operating the software.
- They help in formulating security policy of the software system.

DISADVANTAGES OF NON-FUNCTIONAL REQUIREMENT

- Cons/drawbacks of Non-function requirement are:
- None functional requirement may affect the various high-level software subsystem
- They require special consideration during the software architecture/high-level design phase which increases costs.
- Their implementation does not usually map to the specific software sub-system,
- It is tough to modify non-functional once you pass the architecture phase.

4.2 DESCRIPTION:

Python is a general-purpose programming language that is becoming ever more popular analysing data . Python also lets you work quickly and integrate systems more effectively. Companies from all around d the world is utilizing Python to gather bits of knowledge from their data

Libraries:

First things first, we have to install some libraries so that our program works.

- Here is a list of the libraries we will install: pandas, NumPy, Kera's, and TensorFlow.
- TensorFlow has to be installed so that can work.
- It is an API designed for human beings, not machines.
- Kera's follows best practices for cognitive load: it offers consistent & simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear & actionable error messages.
- It also has extensive documentation and developer guides.

4.3 UML DIAGRAMS:

UML (Unified Modeling Language) is a standard vernacular for choosing, envisioning, making, and specifying the collectibles of programming structures. UML is a pictorial vernacular used to make programming blue prints. It is in like way used to exhibit non programming structures similarly like process stream in a gathering unit and so forth.

4.3.1 USE CASE DIAGRAM:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

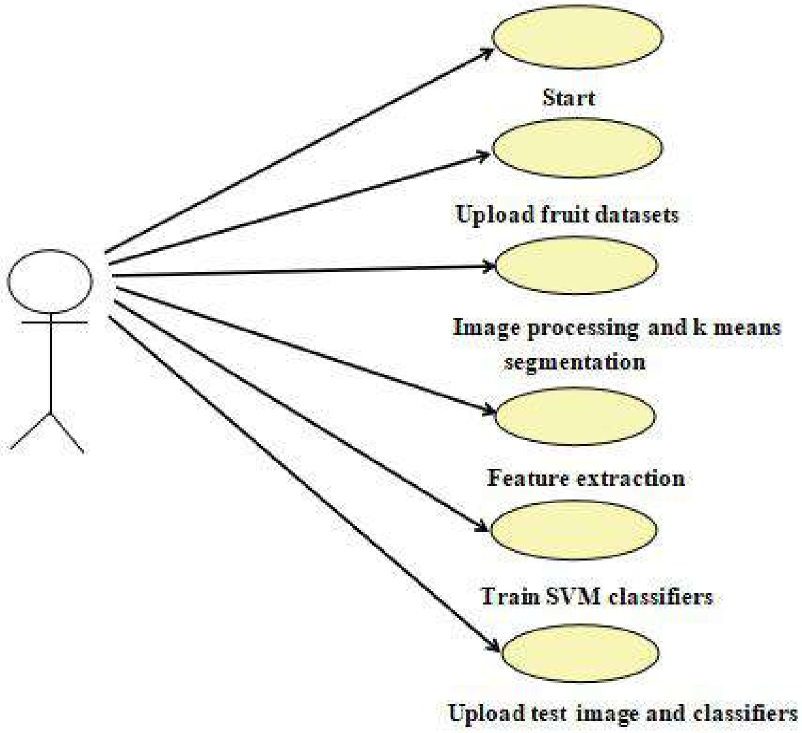

Start
Upload fruit datasets
Image processing and k means segmentation
Feature extraction
Train SVM classifiers
Upload test image and classifiers

Fig 4.3.1: Use case diagram

4.3.2 CLASS DIAGRAM:

Class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. Class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modelling of object oriented systems because they are the only UML diagrams, which can be mapped directly with object-oriented languages. Class diagram shows a collection of classes, interfaces, associations, collaborations, and constraints. It is also known as a structural diagram.

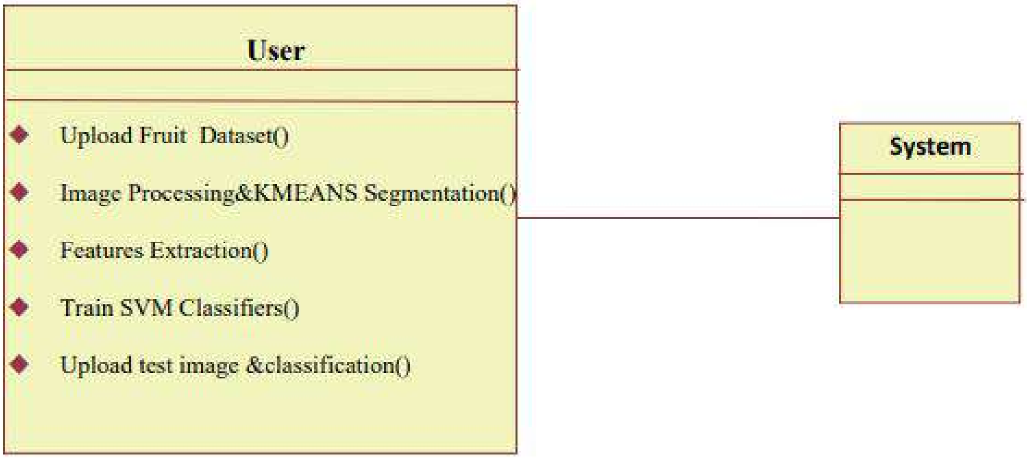


Fig 4.3.2: Class diagram

4.3.3 SEQUENCE DIAGRAM

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

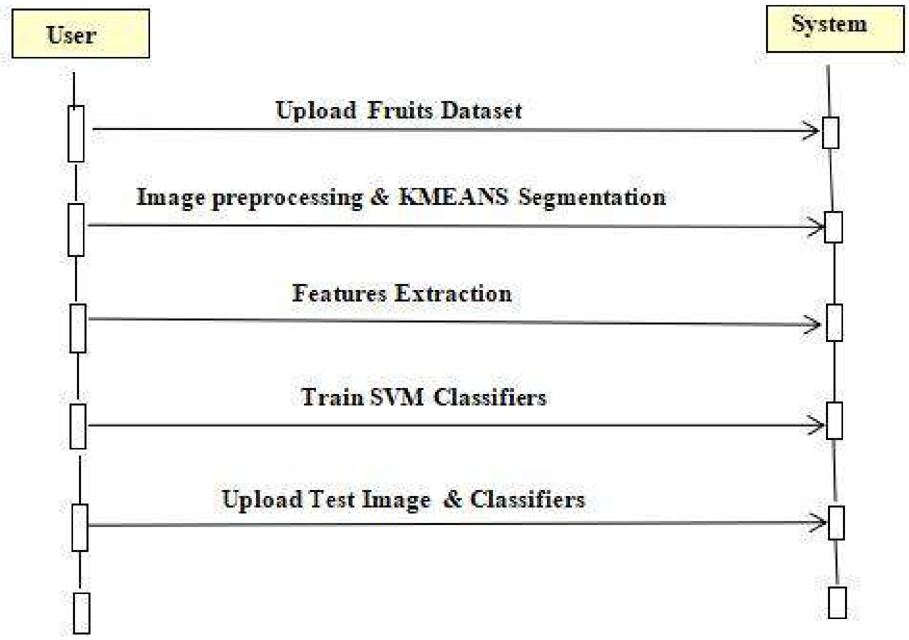

User
System
Upload Fruits Dataset
Image preprocessing & KMEANS Segmentation
Features Extraction
Train SVM Classifiers
Upload Test Image & Classifiers

Fig 4.3.3: Sequence diagram

4.3.4 Flow chart:

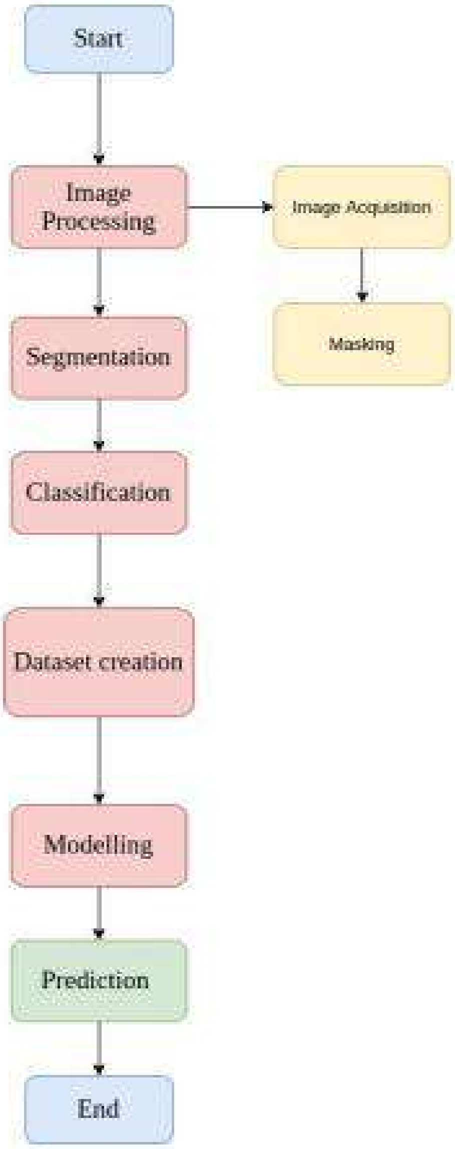


Fig4.3.4: Flow chart

5. IMPLEMENTATION

5.1 SAMPLE CODE

```
from tkinter import messagebox

from tkinter import *

from tkinter import simpledialog

import tkinter

from tkinter import filedialog

import matplotlib.pyplot as plt

from tkinter.filedialog import askopenfilename

from sklearn.model_selection import train_test_split

from sklearn.metrics import accuracy_score

import numpy as np

import pandas as pd

from genetic_selection import GeneticSelectionCV

from sklearn.metrics import classification_report

from sklearn.metrics import confusion_matrix

from sklearn import svm

from keras.models import Sequential

from keras.layers import Dense

import time
```

```
   main = tkinter.Tk()

   main.title("Android Malware Detection")

   main.geometry("1300x1200")

  global filename

   global train

   global svm_acc, nn_acc, svmga_acc, annga_acc

   global X_train, X_test, y_train, y_test

   global svmga_classifier

   global nnga_classifier

   global svm_time,svmga_time,nn_time,nnga_time

def upload():

      global filename

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

      pathlabel.config(text=filename)

      text.delete('1.0', END)

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

 def generateModel():

      global X_train, X_test, y_train, y_test

      text.delete('1.0', END)

      train = pd.read_csv(filename)
```

```
    rows = train.shape[0]  # gives number of row count

    cols = train.shape[1]  # gives number of col count

    features = cols - 1

    print(features)

    X = train.values[:, 0:features]



  Y = train.values[:, features]

    print(Y)

    X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size = 0.2,
      random_state = 0)

    text.insert(END,"Dataset Length : "+str(len(X))+"\n");

    text.insert(END,"Splitted Training Length : "+str(len(X_train))+"\n");

    text.insert(END,"Splitted Test Length : "+str(len(X_test))+"\n\n");

def prediction(X_test, cls):  #prediction done here

    y_pred = cls.predict(X_test)

    for i in range(len(X_test)):

      print("X=%s, Predicted=%s" % (X_test[i], y_pred[i]))

    return y_pred

  def cal_accuracy(y_test, y_pred, details):

    cm = confusion_matrix(y_test, y_pred)

    accuracy = accuracy_score(y_test,y_pred)*100

    text.insert(END,details+"\n\n")
```

```
            text.insert(END,"Accuracy : "+str(accuracy)+"\n\n")

            text.insert(END,"Report :
              "+str(classification_report(y_test,y_pred))+"\n")

            text.insert(END,"Confusion Matrix : "+str(cm)+"\n\n\n\n\n")

            return accuracy

        def runSVM():

            global svm_acc

            global svm_time

            start_time = time.time()

            text.delete('1.0', END)

            cls = svm.SVC(C=2.0,gamma='scale',kernel = 'rbf', random_state = 2)

            cls.fit(X_train, y_train)

            prediction_data = prediction(X_test, cls)

            svm_acc = cal_accuracy(y_test, prediction_data,'SVM Accuracy')

            svm_time = (time.time() - start_time)

def runSVMGenetic():

            text.delete('1.0', END)

            global svmga_acc

            global svmga_classifier

            global svmga_time

            estimator = svm.SVC(C=2.0,gamma='scale',kernel = 'rbf', random_state
         = 2)
```

```
svmga_classifier = GeneticSelectionCV(estimator,
                              cv=5,
                              verbose=1,
                              scoring="accuracy",
                              max_features=5,
                              n_population=50,
                              crossover_proba=0.5,

                             mutation_proba=0.2,
                              n_generations=40,
                              crossover_independent_proba=0.5,
                              mutation_independent_proba=0.05,
                              tournament_size=3,
                              n_gen_no_change=10,
                              caching=True,
                              n_jobs=-1)
start_time = time.time()
svmga_classifier = svmga_classifier.fit(X_train, y_train)
svmga_time = svm_time/2
prediction_data = prediction(X_test, svmga_classifier)
svmga_acc = cal_accuracy(y_test, prediction_data,'SVM with GA
Algorithm Accuracy, Classification Report & Confusion Matrix')
```

```
    def runNN():

        global nn_acc

        global nn_time

        text.delete('1.0', END)

        start_time = time.time()

        model = Sequential()

        model.add(Dense(4, input_dim=215, activation='relu'))

        model.add(Dense(215, activation='relu'))


    model.add(Dense(1, activation='sigmoid'))

        model.compile(loss='binary_crossentropy', optimizer='adam',
        metrics=['accuracy'])

        model.fit(X_train, y_train, epochs=50, batch_size=64)

        _, ann_acc = model.evaluate(X_test, y_test)

        nn_acc = ann_acc*100

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

        nn_time = (time.time() - start_time;

def runNNGenetic():

        global annga_acc

        global nnga_time

        text.delete('1.0', END)

        train = pd.read_csv(filename)
```

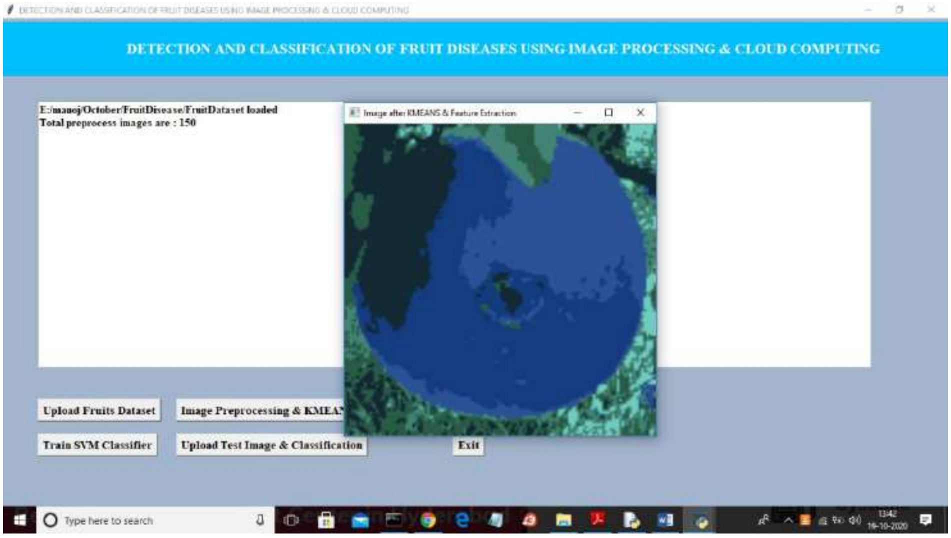
```
  rows = train.shape[0]  # gives number of row count
  cols = train.shape[1]  # gives number of col count
  features = cols - 1
  print(features)
  X = train.values[:, 0:100]
  Y = train.values[:, features]
  print(Y)
  X_train1, X_test1, y_train1, y_test1 = train_test_split(X, Y, test_size =
0.2, random_state = 0)

 model = Sequential()
  model.add(Dense(4, input_dim=100, activation='relu'))
  model.add(Dense(100, activation='relu'))
  model.add(Dense(1, activation='sigmoid'))
  model.compile(loss='binary_crossentropy', optimizer='adam',
  metrics=['accuracy'])
  start_time = time.time()
  model.fit(X_train1, y_train1)
  nnga_time = (time.time() - start_time)
  _, ann_acc = model.evaluate(X_test1, y_test1)
  annga_acc = ann_acc*100
```

```
        text.insert(END,"ANN with Genetic Algorithm Accuracy :
        "+str(annga_acc)+"\n\n")

def graph():

        height = [svm_acc, nn_acc, svmga_acc, annga_acc]

        bars = ('SVM Accuracy','NN Accuracy','SVM Genetic Acc','NN Genetic
      Acc')

        y_pos = np.arange(len(bars))

        plt.bar(y_pos, height)

        plt.xticks(y_pos, bars)

        plt.show()

def timeGraph():

        height = [svm_time,svmga_time,nn_time,nnga_time]


      bars = ('SVM Time','SVM Genetic Time','NN Time','NN Genetic Time')

        y_pos = np.arange(len(bars))

        plt.bar(y_pos, height)

        plt.xticks(y_pos, bars)

        plt.show()

      font = ('times', 16, 'bold')

      title = Label(main, text='Android Malware Detection Using Genetic
      Algorithm based Optimized Feature Selection and Machine Learning')

      #title.config(bg='brown', fg='white')
```

```
    title.config(font=font)
    title.config(height=3, width=120)
    title.place(x=0,y=5)
    font1 = ('times', 14, 'bold')
    uploadButton = Button(main, text="Upload Android Malware Dataset",
    command=upload)
    uploadButton.place(x=50,y=100)
    uploadButton.config(font=font1)
    pathlabel = Label(main)
    pathlabel.config(bg='brown', fg='white')
    pathlabel.config(font=font1)
    pathlabel.place(x=460,y=100)
    generateButton = Button(main, text="Generate Train & Test Model",
    command=generateModel)


    generateButton.place(x=50,y=150)
    generateButton.config(font=font1)
svmButton = Button(main, text="Run SVM Algorithm", command=runSVM)
    svmButton.place(x=330,y=150)
    svmButton.config(font=font1)
    svmgaButton = Button(main, text="Run SVM with Genetic Algorithm",
    command=runSVMGenetic)
```

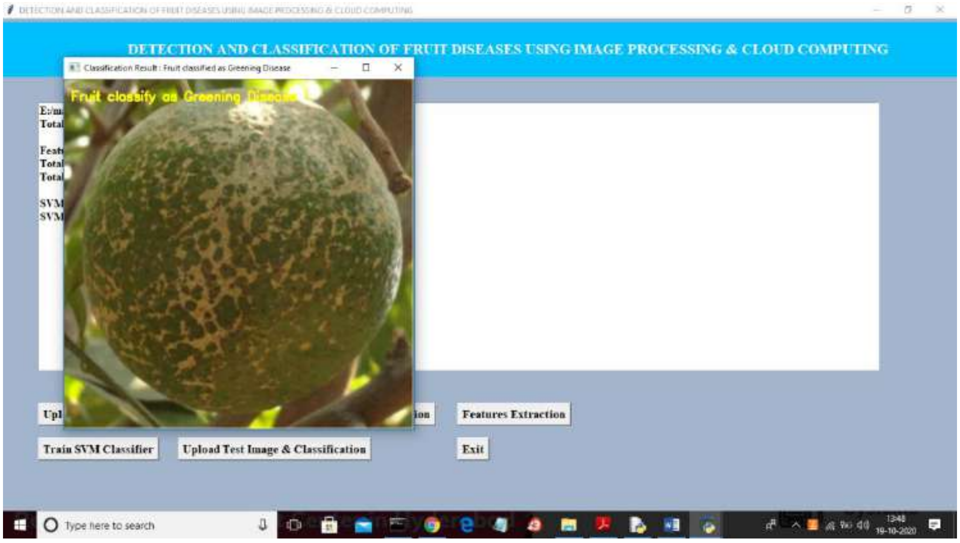
```
svmgaButton.place(x=540,y=150)

svmgaButton.config(font=font1)

nnButton = Button(main, text="Run Neural Network

Algorithm",command=runNN)

nnButton.place(x=870,y=150)

nnButton.config(font=font1)

nngaButton = Button(main, text="Run Neural Network with
GeneticAlgorithm", command=runNNGenetic)

nngaButton.place(x=50,y=200)

nngaButton.config(font=font1)

graphButton = Button(main, text="Accuracy Graph", command=graph)

graphButton.place(x=460,y=200)

graphButton.config(font=font1)

exitButton = Button(main, text="Execution Time Graph",


command=timeGraph)

exitButton.place(x=650,y=200)

exitButton.config(font=font1)

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

text=Text(main,height=20,width=150)

scroll=Scrollbar(text)

text.configure(yscrollcommand=scroll.set)
text.place(x=10,y=250)

text.config(font=font1)

#main.config()

main.mainloop()
```

6. SCREEN SHOTS


DETECTION AND CLASSIFICATION OF FRUIT DISEASES USING IMAGE PROCESSING & CLOUD COMPUTING
DETECTION AND CLASSIFICATION OF FRUIT DISEASES USING IMAGE PROCESSING & CLOUD COMPUTING
Image after KMEANS & Feature Extraction
Upload Fruits Dataset
Image Preprocessing & KMEA
Train SVM Classifier
Upload Test Image & Classification
Exit
Type here to search
13:42
16-10-2020

E:/manoj/October/FruitDisease/FruitDataset loaded
Total preprocess images are : 150

SCREEN SHOT 6.1: Scanning of fruit

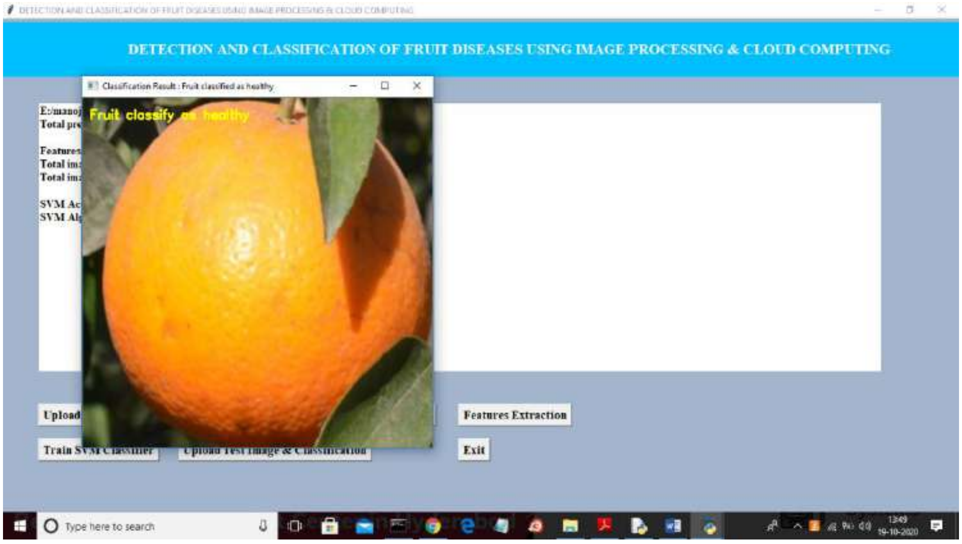

DETECTION AND CLASSIFICATION OF FRUIT DISEASES USING IMAGE PROCESSING & CLOUD COMPUTING
DETECTION AND CLASSIFICATION OF FRUIT DISEASES USING IMAGE PROCESSING & CLOUD COMPUTING
Classification Result : Fruit classified as Greening Disease
Fruit classify as Greening Disease
Upl
ion
Features Extraction
Train SVM Classifier
Upload Test Image & Classification
Exit
Type here to search
13:48
19-10-2020

E:/m
Total

Feat
Total
Total

SVM
SVM

SCREEN SHOT 6.2: Greening disease



DETECTION AND CLASSIFICATION OF FRUIT DISEASES USING IMAGE PROCESSING & CLOUD COMPUTING

Classification Result : Fruit classified as healthy

Fruit classify as healthy

```
E:/manoj
Total pr

Features
Total im
Total im

SVM Ac
SVM Al
```

DETECTION AND CLASSIFICATION OF FRUIT DISEASES USING IMAGE PROCESSING & CLOUD COMPUTING
Upload
Features Extraction
Train SVM Classifier
Upload Test Image & Classification
Exit
Type here to search
13:49
19-10-2020

SCREEN SHOT 6.3: Healthy fruit

7. TESTING

7.1 SOFTWARE TESTING

Testing

Testing is a process of executing a program with the aim of finding error. To make our software perform well it should be error free. If testing is done successfully it will remove all the errors from the software

7.1.1 Types of Testing

1. White Box Testing
2. Black Box Testing
3. Unit testing
4. Integration Testing
5. Alpha Testing
6. Beta Testing
7. Performance Testing and so on

White Box Testing

Testing technique based on knowledge of the internal logic of an application's code and includes tests like coverage of code statements, branches, paths, conditions. It is performed by software developers

Black Box Testing

A method of software testing that verifies the functionality of an application without having specific knowledge of the application’s code/internal structure tests are based on requirements and functionality.

Unit Testing

Software verification and validation method in which a programmer tests if individual units of source code are fit for use. It is usually conducted by the development team.

IntegrationTesting

The phase in software testing in which individual software modules are combined and tested as a group. It is usually conducted by testing teams.

Alpha Testing

Type of testing a software product or system conducted at the developer's site. Usually it is performed by the end users.

Beta Testing

Final testing before releasing application for commercial purpose. It is typically done by end- users or others.

Performance Testing

Functional testing conducted to evaluate the compliance of a system or component with specified performance requirements. It is usually conducted by the performance engineer.

Black Box Testing

Blackbox testing is testing the functionality of an application without knowing the details of its implementation including internal program structure, datastructures etc. Test cases for black box testing are created based on the requirement specifications. Therefore, it is also called as specification-based testing. Fig.7.1 represents the black box testing:

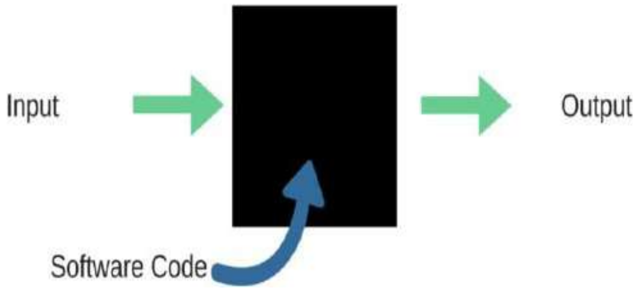


FIGURE 7.1: Black Box Testing

When applied to machine learning models, black box testing would mean testing machine learning models without knowing the internal details such as features of the machine learning model, the algorithm used to create the model etc. The challenge, however, is to verify the test outcome against the expected values that are known beforehand.

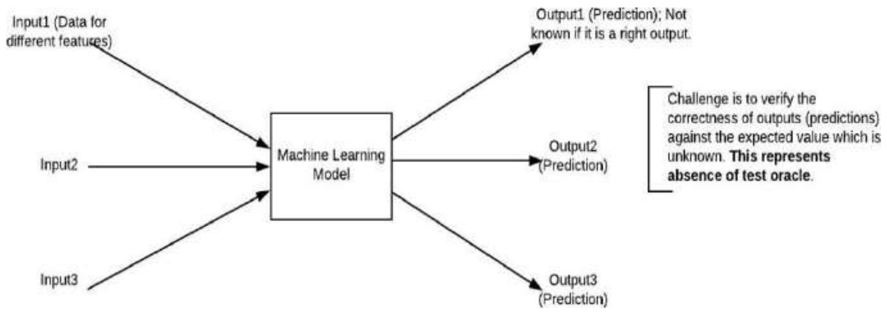

Input1 (Data for different features)
Input2
Input3
Machine Learning Model
Output1 (Prediction); Not known if it is a right output.
Output2 (Prediction)
Output3 (Prediction)
Challenge is to verify the correctness of outputs (predictions) against the expected value which is unknown. This represents absence of test oracle.

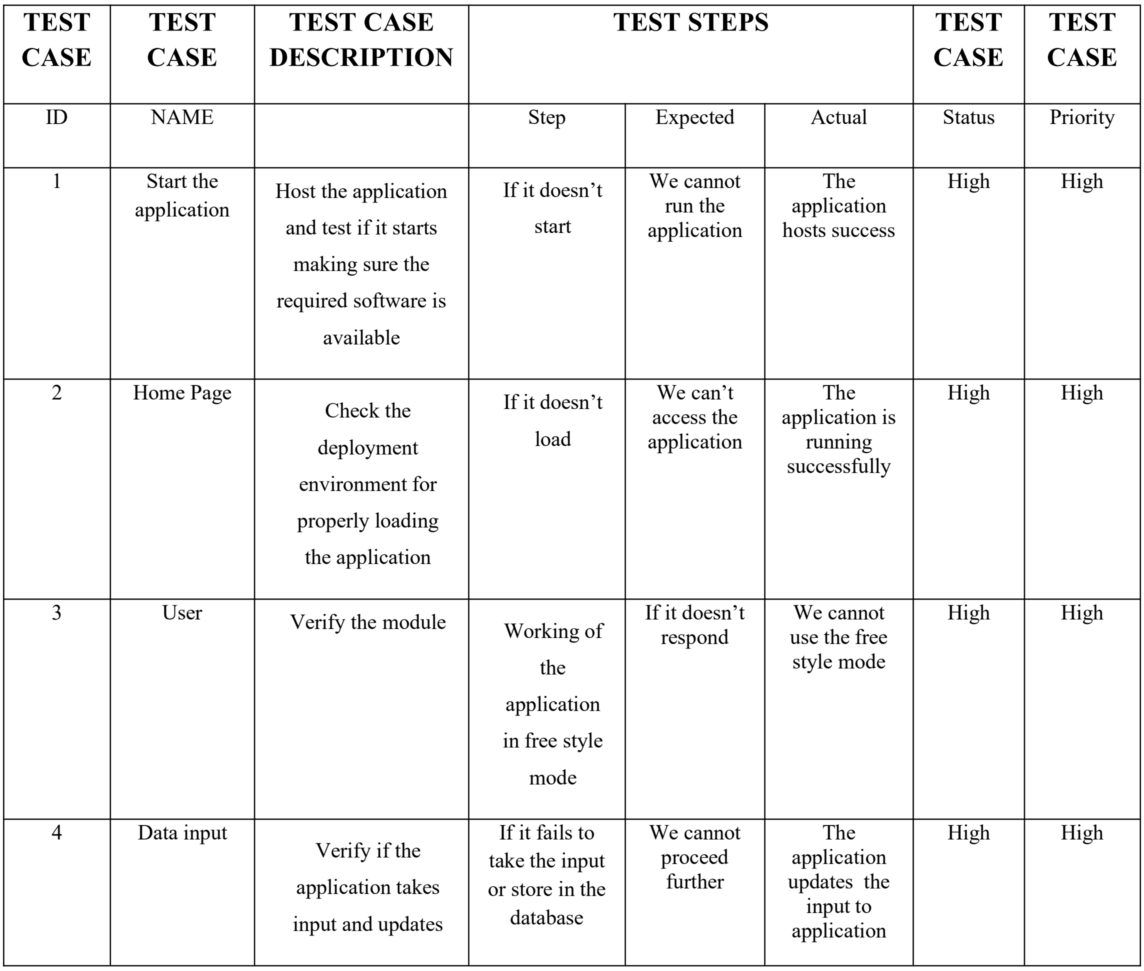
FIGURE 7.2: Black Box Testing for Machine Learning algorithms

The above Fig.7.2 represents the black box testing procedure for machine learning algorithms.

INPUT	ACTUAL OUTPUT	PREDICTED OUTPUT
[16,6,324,0,0,0,22,0,0,0,0,0,0]	0	0
[16,7,263,7,0,2,700,9,10,1153,832,9,2]	1	1

FIGURE 7.3: Black box testing table

The above Fig 7.3 represents the black box testing procedure for machine learning algorithms.



TEST CASE	**TEST CASE**	**TEST CASE DESCRIPTION**	**TEST STEPS**			**TEST CASE**	**TEST CASE**
ID	NAME		Step	Expected	Actual	Status	Priority
1	Start the application	Host the application and test if it starts making sure the required software is available	If it doesn't start	We cannot run the application	The application hosts success	High	High
2	Home Page	Check the deployment environment for properly loading the application	If it doesn't load	We can't access the application	The application is running successfully	High	High
3	User	Verify the module	Working of the application in free style mode	If it doesn't respond	We cannot use the free style mode	High	High
4	Data input	Verify if the application takes input and updates	If it fails to take the input or store in the database	We cannot proceed further	The application updates the input to application	High	High

FIGURE 7.4: Model of test cases

The model gives out the correct output when different inputs are given which are mentioned in Table 7. 4 Therefore the program is said to be executed as expected or correct program

8. CONCLUSION & FUTURE SCOPE

8.1 CONCLUSION:

The development of cloud based scheme for helping Indian farmers and agriculture, helps to analyze the agriculture data in a better way to reduce the hoardings and in bringing up a prosperous safe and peaceful farmer society in India. The classification and segmentation of fruit images were performed using K-Means Algorithm and SVM technique. The various features of few fruits were initially extracted and segment the respective images. After comparison with feature values, the various disease names are analyzed and the optimal disease for the image is identified and the disease is indicated by an alert box and can be provided as the message through mobile application. The total number of samples provided, the true and false positions, the true and false negativities, the accuracy and the specificity are also indicated in an alert box.

8.2 FUTURESCOPE:

In enhancement we will add some ML Algorithms to increase accuracy

9 BIBILOGRAPHY

REFERENCES

[1] Asha R.PatilVarshaI.Patil, B.S.Panchbhai,"Detection of Plant Diseases Using ImageProcessing Tools". Asha R. PatilVarshaI.Patil.Int. Journal of Engineering Research andApplication ISSN : 2248-9622, Vol. 7, Issue 4,(Part -2) April 2017, pp.44-45.

[2] Pratik Agarwal, "AGROCLOUD-Opensurveillance of Indian Agriculture viacloud"2016.International Conference onInformation Technology(InCITe)-The NextGeneration IT Summit.

[3] Athmaja S1, Hanumanthappa M2,"Applications of Mobile Cloud Computing andBig Data Analytics in Agriculture Sector- ASurvey". International Journal of AdvancedResearch in Computer and CommunicationEngineering ICRITCSA M S Ramaiah Instituteof Technology, Bangalore Vol. 5, Special Issue2, October 2016.Authorized licensed use limited to: INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR. Downloaded on August 24,2020 at 22:59:23 UTC from IEEE Xplore. Restrictions apply.

[4] Wang-Su Jeon1 and Sang-Yong Rhee,"PlantLeaf Recognition Using a Convolution NeuralNetwork"International Journal of Fuzzy Logicand Intelligent Systems Vol. 17, No. 1, March2017, pp. 26-34

[5] Li Tan1,2†,Hongfei Hou1,Qin Zhang2."AnExtensible Software Platform for Cloud-basedDecision Support and Automation in PrecisionAgriculture".2016 IEEE 17th InternationalConference on Information Reuse andIntegration.

[6] P. Deepan"Detection and Classification ofPlant Leaf Diseases by using Deep LearningAlgorithm"International Journal of EngineeringResearch & Technology (IJERT) ISSN: 2278-0181Published by, www.ijert.org,ICONNECT -2k18

[7] JagadeeshD.Pujari.a, RajeshYakkundimuthb,*,AbdulmunafS.Byadgi.c,“Image Processing Based Detection of FungalDiseases in Plants”. International Conference onInformation and CommunicationTechnologies(ICICT 2014).Procedia ComputerScience 46 (2015) 1802 – 1808.

[8] Shiv Ram Dubey1, Pushkar Dixit2, NishantSingh3, Jay Prakash Gupta4, “Infected Fruit PartDetection Using K-Means ClusteringSegmentation Technique”. International Journalof Artificial Intelligence and InteractiveMultimedia, Vol. 2, Nº 2.