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	**Dr.A.Raji Reddy**
Associate Professor	**DIRECTOR**
INTERNAL GUIDE	

Dr. K. Srujan Raju	**EXTERNAL EXAMINER**
HOD	

Submitted for viva voice Examination held on ________________

ACKNOWLEDGEMENT

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:

Dr. M. Varaprasad Rao, Mr. J. Narasimha Rao, Dr. T. S.Mastan Rao, Dr.Suwarna Gothane, Mr. A. Uday Kiran, Mr. A. Kiran Kumar, 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.

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. 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

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 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..

2. LITERATURE SURVEY

The Indian agriculture desperately is lagging behind in per hectare yield in almost all crops in comparison to other countries with respect to the population that is needed to be fed. The use of technology in agriculture may help in increasing the productivity and may improve the condition of Indian farmers and protection of their product. The major problem of Indian agriculture is of providing information to the farmers and storing the crop related information at some place for analyzing later. The present paper illustrates a scheme of keeping records on the progress of agriculture, its production and farmers in India, via Cloud, exclusively employed for the welfare of the farmer society and the agricultural practices, GDP, and the cloud being open to general public as well, for studies, and process transparency. Agriculture has been the base for society and livelihood of the people. According to an estimate more than 60% of people are dependent on agriculture for their livelihood. The percentage of cultivable land is very high in INDIA, however, we lag behind in productivity. The per hectare yield is well behind the world average in almost all crops most of the times. When you have a population of billions, the advancement in the agriculture is must.. The technology like agri-apps may help the farmers to make better decisions about seed selection, crop rotation, nutrients, pricing and markets. The technology may help farmers and Indian agriculture to come out of present disastrous state. Cloud Computing, or simply "cloud", is the delivery of on demand computer resources, from applications to data centers and servers, over the Internet on a pay-as-you-go basis. This will help to spread awareness to the people, and will help in taking the necessary actions, for the local bodies and/or Government.This motivated us to propose a cloud based scheme to help farmers for better decision making about best farming practices as well as help the government for making better decisions for farmers and Indian agriculture, and keeping a count on the production and a crops-in-market to a greater extent.

I. RELATED WORK:

This scheme uses sensors, cameras and other sophisticated instruments to improve the quantity and quality of productions. The use of sophisticated and costly instruments makes it very costly, especially in developing countries in India where you need to feed over a billion mouths, inviting thefts as well. The scheme is not suited where land holdings are small and farmers are not in a position to deploy such costly instruments. Indian Agriculture requires a scheme which is not too complex to be used by comparatively less educated farmers of India and comparatively economically weaker farmers of India. But it was limited to agriculture advises only. This motivated us to propose a cost effective cloud based scheme which is based upon interaction between state body or an Agriculture-surveillance-Committee representatives and farmers for getting the data to be analyzed centrally at the cloud.

I. STAKE HOLDERS OF SCHEME:

The collection of older records can be the common field between the two schemes. As India is a vast country and most of its land is employed in farming, the cost of installing IT devices will be huge, and repair and replacements will include huge overhead costs.

II. SCHEME:

The Committee/Board appoints a representative(s) for the data collection from the farmers. The representative will work as an interface between farmer and the government. His main job will include:

1. Visit the city/district/sector assigned to him on weekly/monthly basis and to find out he voids that the farmer is facing in farming e.g. the disturbed pH of soil, wrong or harmful or altered quantity of pesticide being used, scarcity of fertilizers, etc.

2. He shall suggest the farmers improved ways for increasing the production and income. It can be modern farming techniques like drip irrigation .

3. Then he would go to his office and he would study, correct, calculate, and upload the data on the cloud, which will generate the required fund to be raised,S and all the required informations would be displayed on the website, to general public.

III. NEED FOR SCHEME

1. The farmer gets nationwide recognition, and his existence is recorded in the database, including his and his family's personal identity documents, if not, they are requested to get processed.

2. Here, the data collected about the farmer’s status, excluding his personal details, is kept open. Like what he sows, the present condition, ratings of crops, location, his financial status, and donation option. Other options can include total production

The objective of this literature survey was to identify the applications of Mobile Cloud Computing and Big Data Analytics techniques in the agriculture sector. Related literature from IEEE journals and other International journals were collected and reviewed. A conclusion is made by proposing a new model that uses mobile cloud computing and big data analytics techniques together to meet several challenges that the farmers are facing today in the agriculture sector. The proposed model helps farmers in making optimal decisions on their agricultural production and thereby reducing the post-harvest wastage of their products. Agriculture plays a vital role in the overall economic and social well-being of any nation. Agricultural development had helped to a greater extent in the process of industrialization in developed countries such as USA, Japan and England. That is access to right information at right time at right place is essential in reducing several agricultural challenges. This is possible by bringing the possibilities of data analytics techniques, cloud computing and mobile technology under one umbrella in the agricultural sector. Application of big data analytics in the agricultural sector to reduce the challenges that the farmers are facing today due to the improper planning of agricultural production is a promising area of research. This survey paper reviews various papers written on big data analytics techniques in collaboration with mobile technology and cloud computing to help farmers and farming communities to improve agricultural productivity and sustainability.

I. THE SCOPE OF MOBILE CLOUD COMPUTING IN AGRICULTURE SECTOR

Mobile and cloud computing techniques have been used in vast and varied application areas such as banking, healthcare, entertainment, agriculture, etc. all around the world nowadays.Mobile devices play a vital role in the farmer‟s daily business by letting them use mobile application and services. Cloud Computing technology makes use of the Internet and central remote servers to store and process data and thereby allows for much more efficient computing process. The centralized cloud applications can be accessed over the wireless connection by a browser on the mobile devices. Even though the client is a mobile device; the main concept is still cloud computing. MCC offers several advantages to agriculture sector and many positive impacts to farmers. Without worrying about the hardware & software investment and the technical knowledge, farmers can send their requests for a specific cloud service using their mobile phones with an Internet connection. The cloud service provider will process the request dynamically, and finally will send back the results to the client. Nowadays the MCC technology is widely used in the agriculture sector. Some of the related works are discussed below. This android app was built to share many updates like different agricultural commodity updates, weather forecast updates, agricultural news updates and also various loan schemes offered by different banks in India to farmers. Krishi Ville app follows client-server paradigm. HTTP Connection is used to send requests and responses. Therefore, the XML data are parsed before presenting to the user on the mobile screen. Various updated loan schemes for many nationalized and agricultural banks are hard coded in the application. Farmers can submit their queries/requests and receive answers from the experts on the spot using their handsets. This model helps farmers to collect information related to farming, weather forecasting, crop analysis, etc. This tool provides some useful advice for young farmers such as information related to crop selection, crop rotation and mixed cropping. This model can also be used by the researchers to identify various species and their diseases. A high-level image processing component is included for crop analysis. Crop images are sent from the client side. Low-level image processing operations such as colour transformation, gamma correction, linear and non-linear filtering, simple noise reduction, image enhancement, etc can be performed on the client mobile devices. These operations are simple and require less memory for execution and can run on mobile devices with a low bit processor. The intermediate-level as well as high-level image processing operations like segmentation, extraction edge, object recognition, etc. are executed using parallel and distributed computing servers. By using the image analysis techniques for disease detection and identification, farmers become aware of the status of their crops. This improves the production rate and also helps farmers to supervise their land in a convenient way. This mobile application was developed using the Android OS version 2.3.5. For simulation, various tools are used; MATLAB 2008a is used for Image Processing, OpenNebula 2.0 toolkits are used on the server machine to manage heterogeneous distributed data. For experiment purpose, the connectivity between mobile client and the cloud is implemented using Wi-Fi. Entire data analysis process is classified into two steps: the pre-processing and machine learning. In the preprocessing part noise removal, normalization, image equalization, and few morphological operations are performed. This app provides several useful insights to farmers like information related to crops, pesticides, insecticides, and financial sector details, etc. Expert planting tips like which crop to plant in summer, which crop to plant in spring, and which crop is suitable for a particular region, etc. are offered. Also information about current agricultural bank loan rates and schemes are provided. In this mobile app, a facility is provided to select a particular crop sowed by the user and once it is done, this app automatically briefs out all the diseases which are susceptible to that crop. An active Internet connection is required for fetching the information. The huge volumes of data are maintained in cloud storage. This application was developed using the latest Android SDK available at Google. Java and Eclipse Juno integrated with ADT plugin are also used. This app includes many screens like splash screen, state selection screen, menu item screen, etc. Previous studies show that mobile phones play a key role in promoting the farmer""s business by providing improved customer relation, enhanced communication with suppliers, extension officers and customers

Plants plays important role in our lives, as it provides us food and oxygen which is needed by all living things. There are thousands of kinds of trees in this world, and it is very difficult to distinguish between them. Experts can however identify by using characteristics of leaves but a common man cannot. Recognition of Plant from images is a challenging computer vision task. Proposed System is deployed to classify Indian local leaves automatically using deep learning. Deep Learning is a self-learning technique used for massive data and recent development in hardware and big data have turned out as a blessing. We have considered 10 different Indian plants leaves. Recognition rate of proposed system is 93%.Plants play a very vital role in our life as they provide us food and oxygen. We employ machine learning because some plants cannot be identified or are not distinguishable except for sometimes when they bloom flowers or bear fruit. Hence, deep learning is used to fulfil our purpose as it can work with large number of images and produces precise and reliable output. We have developed a model in VGG16 for regional leaves which are found in Maharashtra. We use VGG16 to get better results.The of rest the paper is organized as follows, Section I contains the introduction of the importance and use of recognition of a plant, Section II contains the Related work carried out so far, Section III explains the flow andmethodology used in the proposed system, Section IV explains the results and discussion, Section V concludes the work with future work to be carried out. JyotismitaChaki, Ranjan Parekh and Samar Bhattacharya used multi layered approach to identify the plant leaf. This paper represents methods of recognizing heterogeneous leaves by different visual properties.Aparajita Sahay and Min Chen presented a Windows phone application for the identification of leaf species. They worked on a data set of labelled images using weighted KNN.In their application, user can photograph of a single leaf on only white colored background and submit it to the system as an input image. W.H.Rankothge and D.M.S.B Dissanayake presented a plant identification system which is based on image processing and neural network techniques. Image processing techniques including removing noise of image, to normalize leaf area, to reduce the unwanted white background and to scale up image of leaf. Bin Wang and Douglas Brown presented a novel shape descriptor for more accuracy and fast plant leaf identification. They have used well-known Swedish leaf dataset and their Leaf100 dataset. We remove unwanted noise from the image so as to reduce false positive regions from the leaf images. The leaf images contain only one object, the leaf. Since all leaves are not perfectly flat, image capturing would always cast a shadow underneath the leaf. The shadow would disrupt the edge detection as it has a huge contrast with the background, confusing the algorithms to draw the boundary based on shadow instead of on the leaf. Thus, it should be removed before image segmentation. The image RGB value was changed to HSV value. Then, the channel with the clearest contrast between object and shadow was selected and used to identify the object boundary. As HSV value conversion alters the original color, this step serves as guidance for the subsequent edge detection of RGB value leaf images, rather than producing a final image for feature extraction. Cropping is done as it reduces GPU computation and gives faster and better results. 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 [8].The colors of plants are usually green. Moreover, the shades and the variety of changes of water, nutrient, atmosphere and season can cause change of the color, so the color feature has low reliability. The leaves which are common to this belt arahar ,brinjal , cotton , chilly, custard apple, lemon , peanut, mango , tomato ,sweet lime were taken. Out of these 1943 images used for training and remaining for testing. Output of the feature extraction step is feature set. It is used as input of the classifier to recognize the leaf.VGG 16 model was used for training. Proposed system contains single input layer, hidden layers and output layer. Confusion matrix is used for analyzing how well our classifier can recognize samples of different class. A confusion matrix of size k x k associated with a classifier, demonstrates the predicted and actual/target classification, where k is the number of different classes. In confusion matrix we consider only 10 images of each leaves. Accuracy is the percentage of test samples that are correctly recognized by the classifier. Precision is measure of exactness. Recall is measure of completeness.

However, increments in the amount of data might lead to data quality issues, and as these applications scale into big, real-time monitoring systems the problem gets even more challenging. Visualisation is a powerful technique used in these systems that provides an indispensable step in assisting end-users to understand and interpret the data. In this paper, we present a systematic review to synthesise literature related to the use of visualisation techniques in the domain of agriculture. The search identified 61 eligible articles, from which we established end-users, visualisation techniques and data collection methods across different application domains. We found visualisation techniques used in various areas of agriculture, including viticulture, dairy farming, wheat production and irrigation management. Our results show that the majority of DSSs utilise maps, together with satellite imagery, as the central visualisation. Also, we observed that there is an excellent opportunity for dashboards to enable end-users with better interaction support to understand the uncertainty of data. Based on this analysis, we provide design guidelines towards the implementation of more interactive and visual DSSs. In the field of agriculture, different stakeholders such as farmers, advisers and policymakers use software tools that facilitate farm management by gathering data from multiple sources, analysing these data and utilising a series of suggestions that are presented by different visual outputs.Thus, PA research concentrates on enabling users to make the right decisions considering both space and time. Still, models remain unfriendly, inside a black-box and behind DSS software. Visualisation techniques have been used to assist users to better interact and understand data by aggregating, filtering, searching or otherwise sifting through and scaling down relevant information. Moreover, visualisations are often explicitly designed to assist our visual system in handling detail that might otherwise require significant cognitive effort. For instance, visualisations provide information that can be easily perceived, recognised and processed into inferences. In this sense, visualisations also offer short-term or long-term memory aids to reduce memory and cognitive load. Such support can make data, including potentially complex information, more easily consumable. A great number of visualisations and visual analytics tools have been proposed in the domain of agriculture with the aim to support the decision-making process. However, to the best of our knowledge, a comprehensive analysis of such tools, by considering their application areas, visualisation techniques and intendedend-users, does not yet exist. Previous literature reviews in agriculture have focused on similar areas such as DSSs and PA, but not on visualisation techniques.This survey highlighted the opportunities of big data analysis for smarter farming. As such, a number of reviews have addressed the issues surrounding DSSs and PA, but the use of visualisations in DSSs has yet to be understood. To address this gap, in this paper, we review the use of information visualisation techniques in 61 articles that report their support for end-user decision-making. For example, paying little attention to the environment could result in the DSS being unfit for the intended environment. Our second goal was to gain insight into the evaluation of DSS, as well as to learn from these evaluations to identify opportunities for future research. To guide this work, we have defined our research questions as follows:

- RQ1: What visualisation techniques are being used across different domains in agriculture

- RQ2: How are these visualisations being used by end-users to make decision

- RQ3: What is the role of uncertainty in the visualisation tools that support decision-making

- RQ4: What is the role of HCI in the design and development of visualisation tools to support end-user decisions

The contributions of this paper are the following: first, we present an extensive overview of visualisation and visual analytics techniques that have been elaborated in the agriculture field, highlighting the potential of DSSs and their use of visualisations for decision-making in agricultural contexts. Second, we present an analysis of a wide range of characteristics, including visualisation techniques that are commonly used, how data is captured and how these systems have been evaluated. Based on this analysis, we outline guidelines for the design of DSSs as well as directions for future research for this area. In this section, we first elaborate the method employed for the selection of articles/papers for our systematic review.

In this section, we first elaborate the method employed for the selection of articles/papers for our systematic review. We aThis review includes articles that investigated or reported the usage of DSSs involving visualisation techniques to support end-users in the field of agriculture. No restrictions were placed on the date. All the included papers were written in English. Articles describing the use of visualisations in text or figures were included. Papers that reported the design process of the tool and user studies regarding visualisation techniques were included and analysed in detail separately. A scoping exercise to nail down search keywords was conducted in the Google Scholar and ScienceDirect databases in May 2018. The keyword identification was complemented by using keyword services such as Google Keyword Planner and Keyword Tool. The produced keywords are presented in Table Journals in the agriculture domain were identified using ScimagoJR1 . The search of papers within the identified journals was done in the ScienceDirect platform. Results from the search are presented in Table.

Nowadays, overseas commerce has increased drastically in many countries. Plenty fruits are imported from the other nations such as oranges, apples etc. Manual identification of defected fruit is very time consuming. This work presents a novel defect segmentation of fruits based on color features with K-means clustering unsupervised algorithm. We used color images of fruits for defect segmentation. Defect segmentation is carried out into two stages. At first, the pixels are clustered based on their color and spatial features, where the clustering process is accomplished. Then the clustered blocks are merged to a specific number of regions. Using this two step procedure, it is possible to increase the computational efficiency avoiding feature extraction for every pixel in the image of fruits. Although the color is not commonly used for defect segmentation, it produces a high discriminative power for different regions of image. This approach thus provides a feasible robust solution for defect segmentation of fruits. We have taken apple as a case study and evaluated the proposed approach using defected apples. The experimental results clarify the effectiveness of proposed approach to improve the defect segmentation quality in aspects of precision and computational time. The simulation results reveal that the proposed approach is promising. igital images are one of the most key medium of conveying information. Extracting the information from images and understanding them such that the extracted information can be used for several tasks is an important characteristic of Machine learning. Using images for the navigation of robots is an example of the same. Other applications such as extracting malign tissues from the body scans etc form an integral part of Medical diagnosis. Image segmentation is one of the initial steps in direction of understanding images and then finds the different objects in them. Modern agricultural science and technology is extreme advance. The value of fruit depends on the quality of fruit. It is an important issue how to assay quality of fruit in agricultural science and technology. The classical approach of fruits quality assessment is done by the experts and it is very time consuming. Defect segmentation of fruits can be seen as an instance of the image segmentation in which we are interested only to the defected portion of the image. Image segmentation entails the separation or division of the image into areas of similar attributes. In another way, segmentation of the image is nothing but pixel classification. The difficulty to which the image segmentation process is to be carried out mostly depends on the particular problem that is being solved. It is treated as an important operation for meaningful interpretation and analysis of the acquired images. It is one of the most crucial components of image analysis and pattern recognition and still is considered as most challenging tasks for the image processing and image analysis. It has application in several areas like Analysis of Remotely Sensed Image, Medical Science, Traffic System Monitoring, and Fingerprint Recognition and so on. Image segmentation methods are generally based on one of two fundamental properties of the intensity values of image pixels: similarity and discontinuity. In the first category, the concept is to partition the image into several different regions such that the image pixels belonging to a region are similar according to a set of predefined criteria's. Whereas, in the second category, the concept of partition an image on the basis of abrupt changes in the intensity values is used. Edge detection technique is an example of this category which is similar to the boundary extraction. Researchers have been working on these two approaches for years and have given various methods considering those region based properties in mind. But, still, there is no fixed approach for the image segmentation. Based on the discontinuity or similarity criteria, many segmentation methods have been introduced which can be broadly classified into six categories:

(1) Histogram based method,

(2) Edge Detection,

(3) Neural Network based segmentation methods

(4) Physical Model based approach,

Histogram based image segmentation techniques are computationally very efficientwhen compared to other image segmentation techniques because they usually require only a single pass through the image pixels. In this technique, a histogram is calculated from all of the image pixels, and thepeaks and valleys are detected in the histogram. Now the image pixels between two consecutive peaks can be considered to a single cluster. A disadvantage of this method is that it is not able to categorize when the image has no clear gray level histogram peak. Another disadvantage of this method is that the continuity of the segmented image regions cannot be ensured. We should focus on global peaks that are likely to correspond to the dominant image regions for the histogram based segmentation method to be efficient. The edge detection method is very widely used approaches to the image segmentation problems. It works on the basis of the detection of points considering abrupt changes at gray levels. A disadvantage of the edge detection method is that it does not work well when there are many edges in the image because in that case the segmentation technique produces an over segmented output, and it cannot easily identify a boundary or closed curve. For an edge based segmentation method to be efficient, it should identify the global edges and these edges have to be continuous. Neural Network based image segmentation relies on processing small regions of an image using a neural network or a set of different artificial neural networks. After this, the decision-making method marks the regions of an image on the basis of the category recognized by the artificial neural network. Kohonen self organizing map is a type of network designed especially for such type of problems. The physical model based image segmentation technique assumes that for an image, individual regions follow a recurring form of geometrical structure. This type of segmentation methods uses texture feature. The region based image segmentation method uses the similarity of pixels within a region in an image. Sometimes a hybrid method incorporating the region based and edge based methods have been proved to be very useful for some applications. The segmentation method incorporating clustering approaches encounters great difficulties when computing the number of clusters that are present in the feature space or extracting the appropriate feature. This type of image segmentation is widely used due to the simplicity of understanding and more accurate result. This paper presents an efficient image segmentation approach using K-means clustering technique based on color features from the images. Defect segmentation is carried out into two stages. At first, the pixels are clustered based on their color and spatial features, where the clustering process is accomplished. Then the clustered blocks are merged to a specific number of regions. Using this two step procedure, it is possible to increase the computational efficiency avoiding feature extraction for every pixel in the image of fruits. Although the color is not commonly used for defect segmentation, it produces a high discriminative power for different regions of the image. The rest of the paper is organized as follows:

I.A BRIEF OVERVIEW OF RELATED WORK

Color image segmentation has been a difficult task for the researchers over the past two decades Graph cut methods such as normalized cut and minimal cut characterize A major problem for this kind of methods known as the problem of validity is how to decide the number of clusters in any image. Since the problem is basically unresolved, most techniques need that the user should provide a terminating criterion. The soft computing techniques they used were competitive neural network and Possibilistic C means algorithm (PCM). Researchers also used Fuzzy set and Fussy logic techniques for solving segmentation problem. There are various segmentation techniques in medical imaging problems depending on the region of interest in the image. There are region growing segmentation methods and atlas-guided techniques. Some of them use a semi-automatic method and still need some operator relations. Other techniques use fully automatic methods and the operator has just a verification role. They have used fast Entropy thresholding for the extraction of edges. These seeds were then replaced by centroids of the generated homogeneous edge regions by incorporating the additional pixels step by step. This method assigns each pixel in that region as seeds after checking whether the value is under a threshold. They have applied region growing and region merging techniques after the selection of seeds. As discussed above color image segmentation has been widely used by the researchers. In the case of fruit diseases more than one disease may be present at a time so we have to use more than two clusters to segment the infected part with fruit and background.

II. K-MEANS CLUSTERING ALGORITM

The food image processing using clustering is an efficient method. Data partitioning is a usual technique for the analysis of statistical data, which is used in many areas, including machine learning, image analysis, pattern recognition, bioinformatics and data mining. The computational task of partitioning the data set into k subsets is often referred to unsupervised learning. K-means is generally used to determine the natural groupings of pixels present in an image. It is attractive in practice, because it is straightforward and it is generally very fast. It partitions the input dataset into k clusters. Clustering algorithm assumes that a vector space is formed from the data features and tries to identify natural clustering in them. As a part of this work, we implemented an iterative version of K-means algorithm. The algorithm requires a color image as input. The algorithm of K-means clustering is as follows

Step 1 Compute the distribution of the intensity values.

Step 2 Using k random intensities initialize the centroids.

III. DEFECT SEGMENTATION

Image segmentation using k-means algorithm is quite useful for the image analysis. An important goal of image segmentation is to separate the object and background clear regardless the image has blur boundary. Defect segmentation of fruits can be seen as an instance of image segmentation in which number of segmentation is not clearly known. To demonstrate the performance of the proposed approach, we have taken apples as a case study. The introduced method is evaluated on the defected apples. We have taken some of the diseases of the apples such as apple scab, apple rot and apple blotch for the defect segmentation. From the empirical observations it is found that using 3 or 4 clusters yields good segmentation results. So, in this experiment input images are partitioned into three or four segments as per requirement. So, we can say that if defected area is larger, fewer clusters will be required while if defected area is smaller more clusters will be needed. It means number of clusters required for the defect detection from the infected image is invertionally proportion to the defected area. The experimental results suggest that the introduced method for defect segmentation in this paper is robust because it can accurately segment the defected part with the fruit region, background and stem/calyx.

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/Intel
- 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:

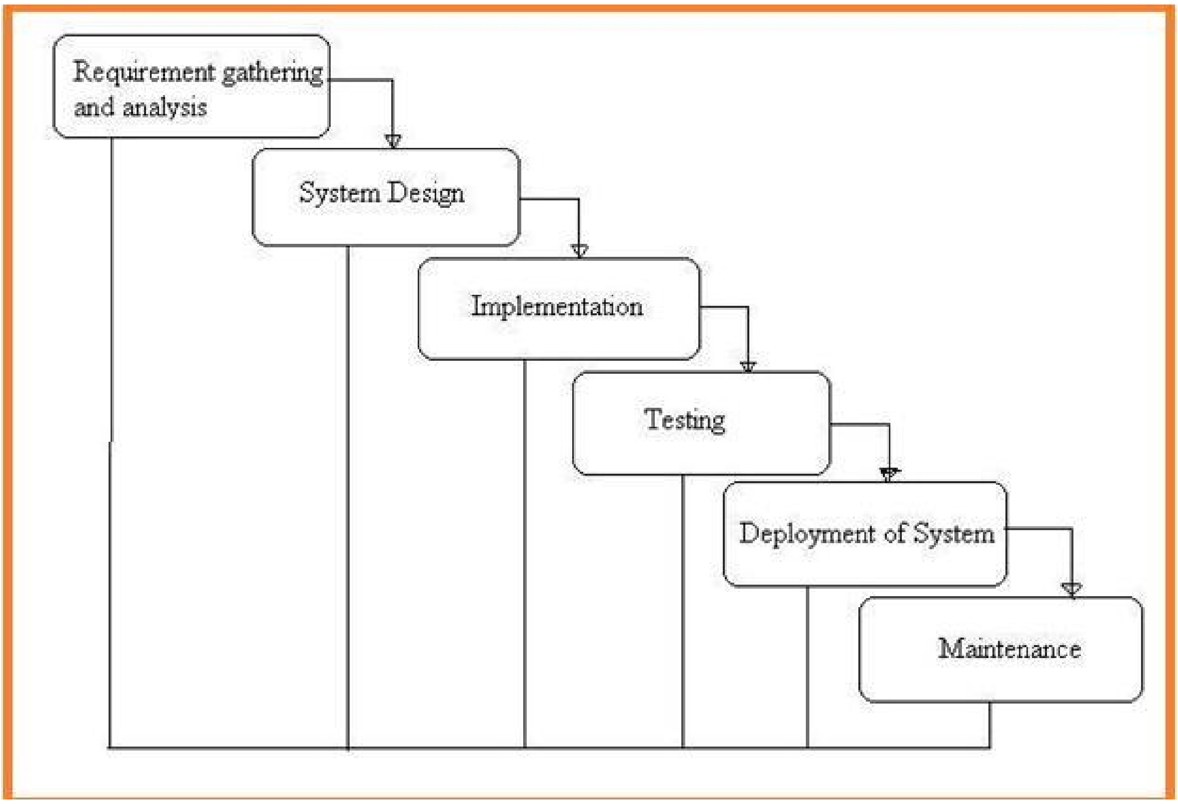


Fig 4.1: Project architecture

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.

4.3UML 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.

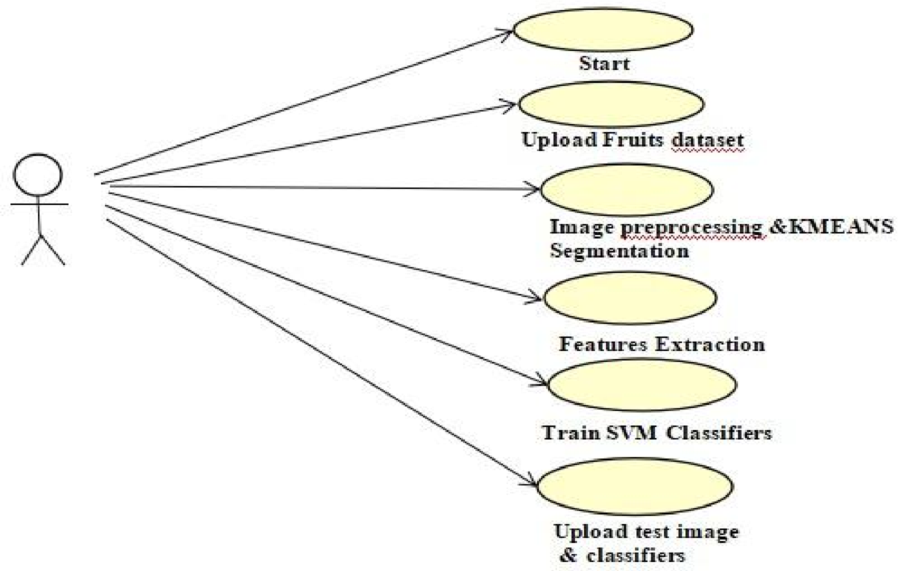


Fig5.1.1: Use case diagram

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.

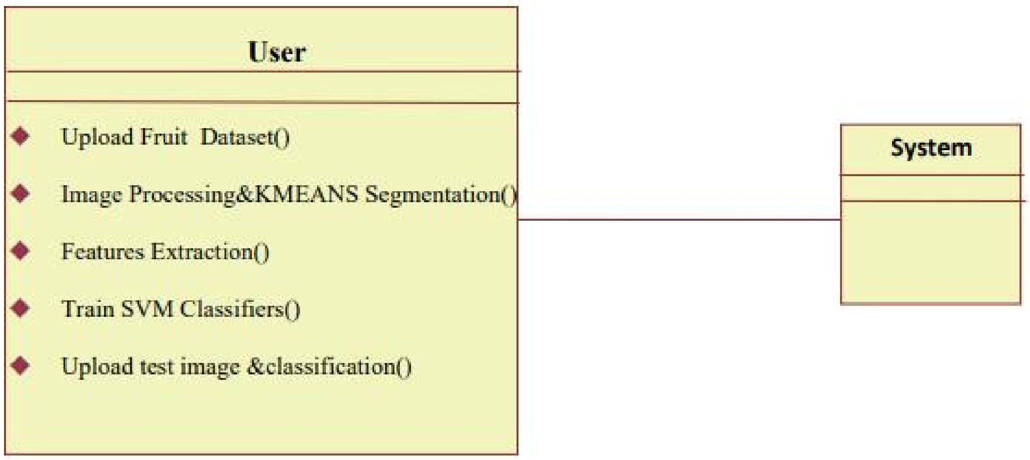


Fig 4.3.2: Class diagram

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.

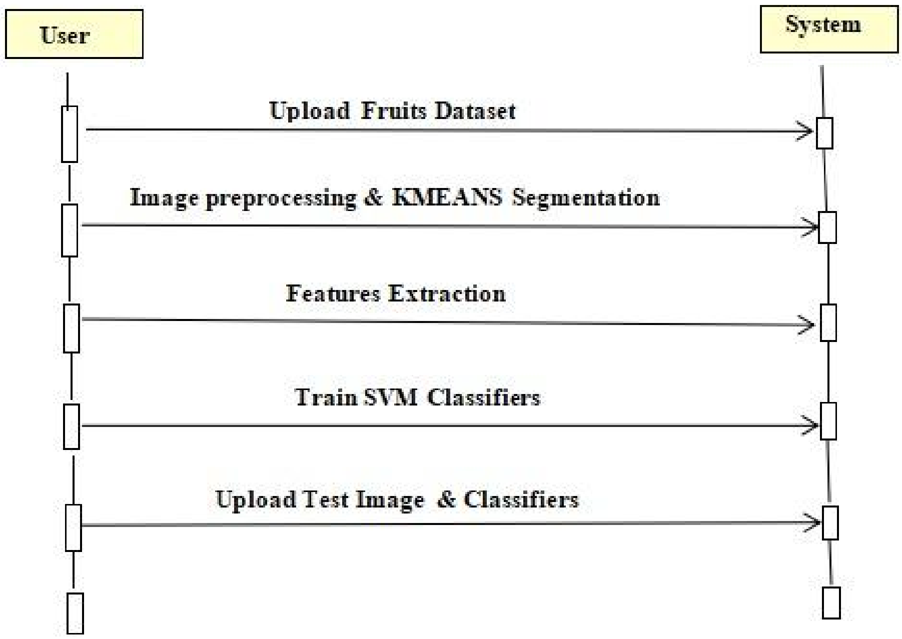


Fig 4.3.3: Sequence diagram

4.3.4 Flow chart:

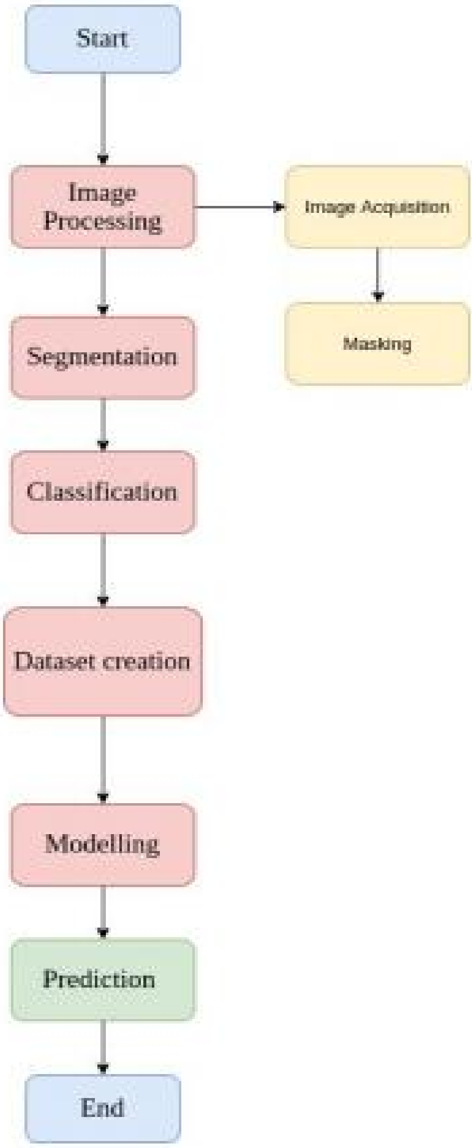


Fig4.3.4: Flow chart

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'))
```

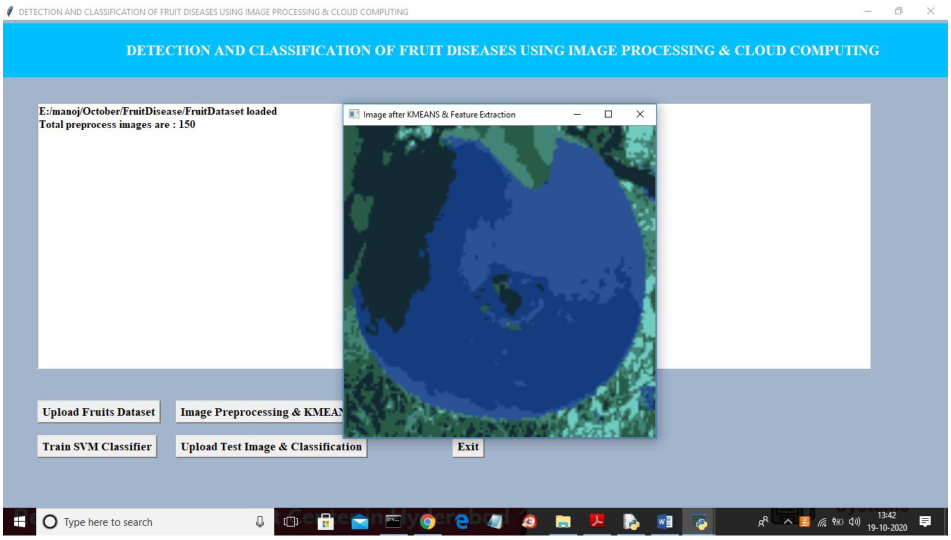
```
        model.add(Dense(100, activation='relu'))

        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)
```

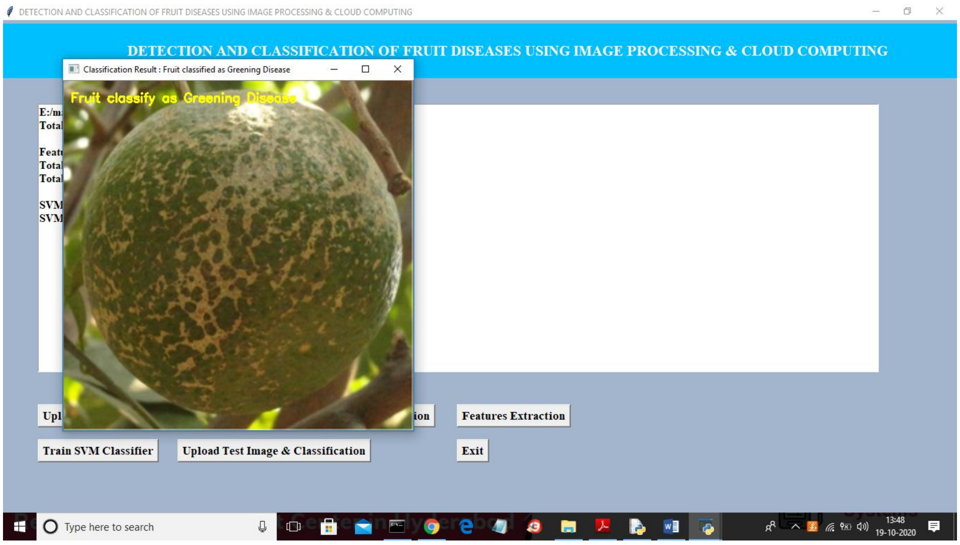
```
svmButton = Button(main, text="Run SVM Algorithm",
command=runSVM)

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 Genetic
Algorithm", 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()
```

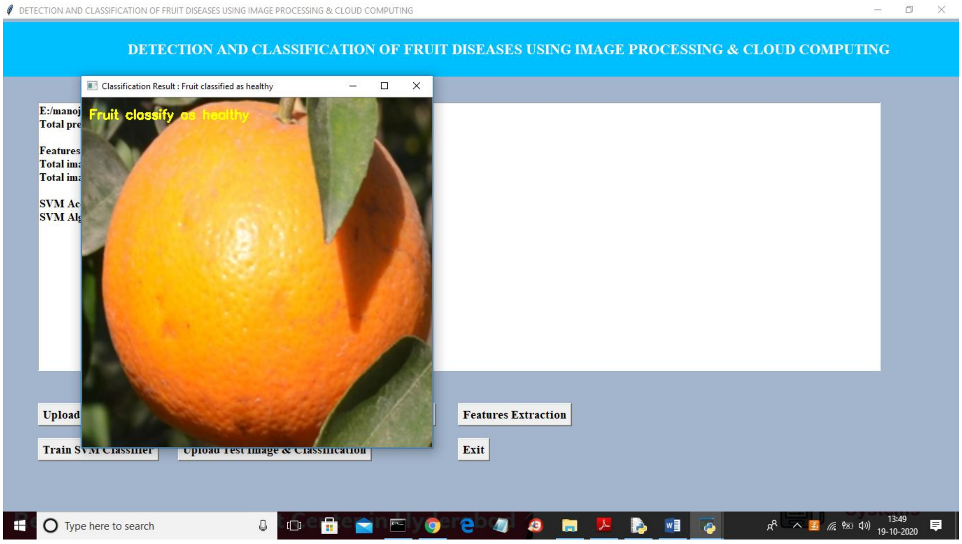
6 .SCREEN SHOTS



SCREEN SHOT 6.1 Scanning of fruit



SCREEN SHOT 6.2 Greening disease



SCREEN SHOT 6.3 Healty 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.

Integration Testing

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, data structures etc. Test cases for black box testing are created based on the requirement specifications. Therefore, it is also called as specification-based testing. Fig.4.1 represents the black box testing:

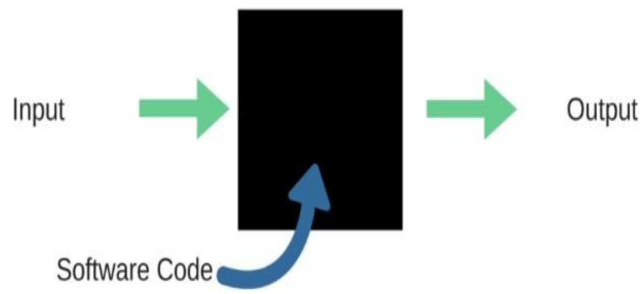


FIGURE 7.1 Black Box Testing

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.

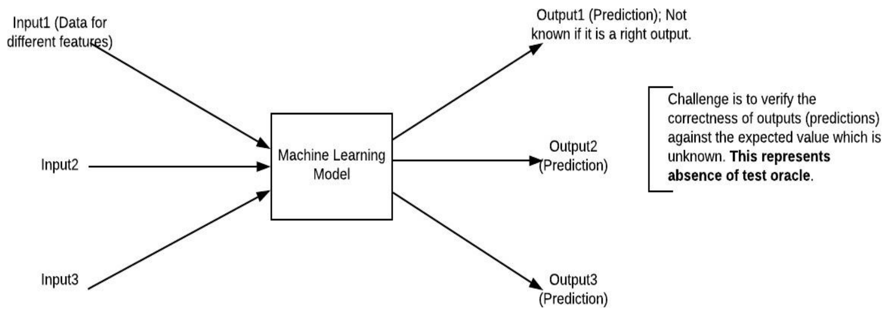


FIGURE 7.2 Black Box Testing for Machine Learning algorithms

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

FIG 7.3 Black box testing table

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

Test Case Id	Test Case Name	Test Case Description	Test Steps			Test Case Status	Test Priority
			Step	Expected	Actual		
01	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
02	Home Page	Check the deployment environment for properly loading the application.	If it doesn't load.	We cannot access the application.	The application is running successfully.	High	High
03	User	Verify the	If it	We	The	High	High

	Mode	working of the application in freestyle mode	doesn't Respond	cannot use the Freestyle mode.	application displays the Freestyle Page		
04	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

Fig 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

6.1 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.

6.2 FUTURE SCOPE:

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-based Decision Support and Automation in PrecisionAgriculture”.2016 IEEE 17th International Conference on Information Reuse and Integration.

[6] P. Deepan“Detection and Classification of Plant Leaf Diseases by using Deep Learning Algorithm”International Journal of Engineering Research & Technology (IJERT) ISSN: 2278- 0181Published by, www.ijert.org,ICONNECT -2k18

[7] JagadeeshD.Pujari.a, Rajesh Yakkundimuthb,*,AbdulmunafS.Byadgi.c,“Image Processing Based Detection of Fungal Diseases in Plants”. International Conference

onInformation and Communication Technologies(ICICT 2014).Procedia ComputerScience 46 (2015) 1802 – 1808.

[8] Shiv Ram Dubey1, Pushkar Dixit2, NishantSingh3, 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.