



Machine Learning System for Detection and Classification of Some Diseases in Selected Plants

Usman, Ali¹., Dr. Joshua Abah²., Falmta B. Wakil¹., Kalli, B. Mai³

¹Department of Computer Engineering, Ramat Polytechnic Maiduguri, Nigeria

²Department of Computer Engineering, University of Maiduguri, Nigeria

³Department of Electrical and Electronic Engineering, Ramat Polytechnic Maiduguri, Nigeria

Abstract: *The system will assist the farmer without formal training to diagnose plant disease through Mobile phone device. The user will start by capturing the image of the defected plant leaf using high resolution mobile phone camera and then sent the image to the server by selecting the options provided on the application. Then the MATLAB fetch the captured image from the server and apply for pre-processing for image enhancement. Segmentation of Captured leaf images is perform using k-means clustering algorithms to form clusters. Features of plant leaves are selected and extracted before applying the clustering algorithms and classification algorithm (Support Vector Machine classification algorithms) for testing and classification by comparing with pre-defined training images stored in the database. Finally, the classification result will then be sent back to user through the application on the phone.*

Key words: *k-means, leaf image, disease, farmers, Classification*

1.0 INTRODUCTION

The cost of agricultural products may increase significantly if diseases are allowed to infest plants. When these occur and are left unchecked, it would negatively affect plant's yield in terms of both quality and quantity hence, constant monitoring of plants is required in order to detect early signs of diseases before it spread Nikos (2017). Farmers loses large amount of money annually and consumers and the end of the value chain buy food products at high prices because of unchecked plant disease infestation. In view of this, there is hope that this research will be of benefit to farmers, consumers of agricultural products and also researchers who may wish to research or elaborate on their work in the future.

2.0 RELATED WORKS

Sandesh & Amit (2017) designed a system called "plant disease detection in image processing using MATLAB". The system which used k-means clustering for segmentation

and Artificial Neural Network techniques for training and classification that detect disease in both leaves and fruits with high accuracy but is very slow, and it can only detect the plant's leaves images saved in the MATLAB database.

In the work of Trimi & Sushma (2017) titled "Plant Disease Detection Using Different Algorithms" and "Artificial Neural Network (feed forward back propagation) for classification" used two different segmentation techniques thresholding and K-means clustering algorithm for segmentation. The system is fast but has limitation as it misclassifies some images which give low accuracy and can only work on MATLAB software only. The papers do not provide a clear comparison with other classification techniques.

Xin, 2017 present the application of machine learning in plant resistance genes discovery and plant diseases classification. The system used Support Vector Machine for classification which provides powerful tool to analyze tremendous amount of data and more machine learning based tools are needed to predict important plant resistance genes. (Bhog & Pawar, 2016) adopt the concept of neural network for the classification and K-means clustering for segmentation in cotton leaf disease analysis (Yellow spot, Red spot, white spot, Cercospora and Alternaria) which is able to achieved recognition accuracy of 89.56% and execution time of 436.95 second using Euclidean distance in MATLAB.

The absence of the labour-intensive phase of feature engineering and the general ability of the solution makes them a very promising candidate for a practical and scalable approach for computational inference of plant diseases in Sharada, David & Marcel (2016). The system used deep convolutional Neural Network for the training to demonstrate the technical feasibility of a deep learning approach to enable automatic disease diagnosis through image recognition. The Overall accuracy of 31.40% obtained is poor.

Only 75.9% of accuracy is achieved in the work of Suhaili et al. (2013) to classify Downey Mildew and Anthracnose for watermelon leaf diseases. Based on RGB color component, region of interest is identifying that is infected from the given testing plant leaf. The work incorporated the concept of Neural network pattern recognition toolbox is used for the classification, image pre-processing and segmentation median filter is used.

Jha, Bababe & Ashok (2017) Design a system called "Plant Disease Information Generating System Based on Android Application Technology" which uses HIS and k-mean for segmentation and clustering, Color Co-occurrence Method for feature extraction and Artificial Neural Network as classification Algorithms. The system is able to classify plant leaves with average percentage accuracy 73.648% which is too low and the researcher uses very few datasets for the experiment.

2.1 LIMITATION OF PREVIOUS RELATED WORK:

Optimization is needed for the previous work to obtain high percentage accuracy; as such best feature extraction method is needed. More information about the plant's leaf such as colour and texture is needed for segmentation as such the extension of database is required to obtain high accuracy.

The researchers were able to cover very few diseases in their work. So, more work needed to cover more diseases. One of the main reason of machine learning misclassification is because disease symptoms varies from one plant to another as such optimization of more plant features is needed for the training dataset to cover more disease cases and to obtain high result percentage accuracy.

3.0 METHODOLOGY

This research adopts a machine learning approach as the core of the system. Here, both unsupervised and supervised machine learning algorithms are employed. The k-means clustering algorithm which is an unsupervised machine learning algorithm is used to prepare and group the input data during the segmentation phase. After the segmentation phase, the support vector machine algorithm which is a supervised machine learning algorithm is then used to carry out the task of classification.

The machine learning approach to image processing is divided into five stages as follows;

- i. Image capturing and acquisition;
- ii. Image pre-processing and segmentation;
- iii. Feature extraction;
- iv. Training and Classification;
- v. Reporting and evaluation.

3.1 IMAGE CAPTURING AND ACQUISITION

Image acquisition is the first step of any digital image processing and it is described as getting the plant leaf image (by snapping plant leafs image using high pixel camera from mobile phone device or any camera device) as shown in Figure 3.2 and sent to MATLAB for processing and classification. It is also the process of retrieving image from the mobile device for further processing. In this work, the healthy plant leaf and the diseased plant leaf images will be capture and stored in the server for MATLAB image processing system, this is called normality model.

This is done by interconnecting the mobile phone to the computer based server, so that the user can sent the captured image to the server for upward forwarding to the MATLAB for processing. The interfaced of the mobile phone is design in such a way that the user can conveniently use it with little training as shown in fig 3.1 below.



Figure 3. 1: Graphical Interface of Android Application

The first sub-system is the Android application installed on the android mobile phone as the system client that provides user with the interface to interact with the functions of the application. The user can choose to take a plant leaf picture from the mobile phone camera or select an image of the plant leaf from the gallery of the mobile phone to be diagnosed. Once the user selects or snap the image and press upload button, the android application will send the image to the web application server via HTTP protocol. Web application server sub-system is implemented with Java Servlet and Tomcat

3.2 IMAGE PRE-PROCESSING AND SEGMENTATION

The main purpose of image pre-processing is to improve the quality of image data by removing unwanted contortions. Pre-processing method uses various techniques such as changing image size and shape, percolate of noise, image conversion, improving image and morphological operations. Various MATLAB code to resize image, to enhance contrast and RGB to grayscale conversion is use for this work as shown in figure 3.2 for further operations like creating clusters in segmentation.



Figure 3. 2: Contrast enhanced

Image segmentation is the method of transforming digital image into several segments and rendering the image into a form that can be processed by the machine learning classifier. The main use of image segmentation is for locating the objects and bounding line of that image. K-means clustering method partitions the images into clusters so that at least one part of the partition (cluster) will contain the image with major area of diseased part.

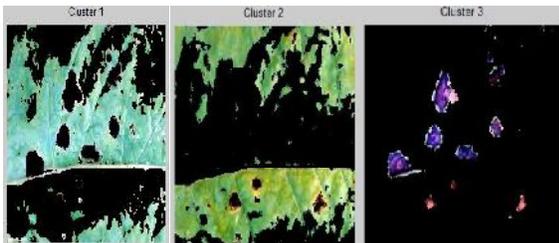


Figure 3. 3: Diseased plant leaf image clusters

3.3 FEATURE EXTRACTION

For this research, *Color Co-occurrence* (CCM) method is used in extracting features. CCM is a method in which both the color and texture of an image are taken into account, to arrive at unique features, which represent that image.

3.4 BLOCK DIAGRAM DESCRIBING THE APPLICATION OF THE WHOLE PROCESS

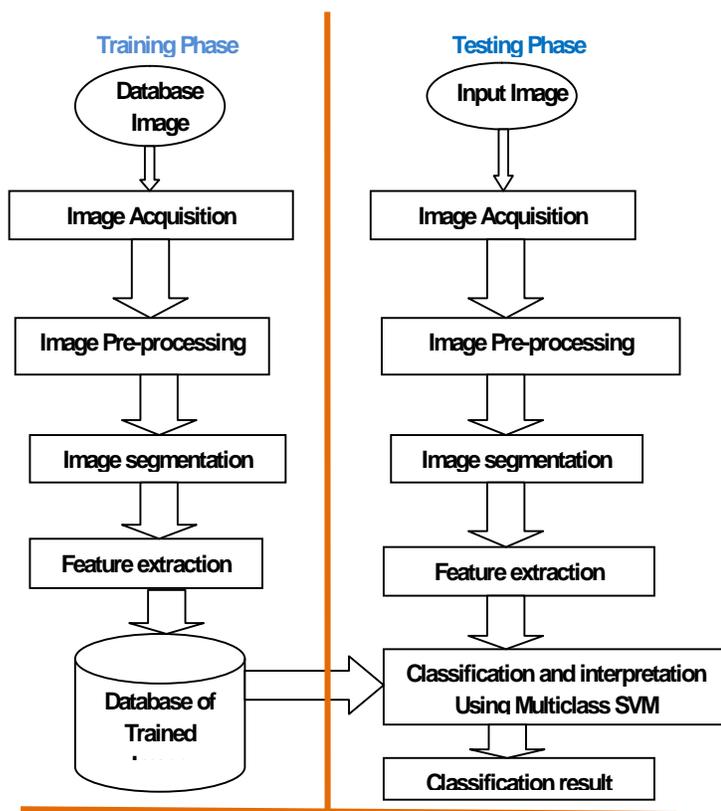


Figure3.4: Fundamental Steps of Digital Image Processing

4.1 ANDROID MOBILE PHONE APPLICATION IMPLEMENTATION

The important of this system is the ability of farmers without formal trainings to diagnose plant diseases in order to increase crop production, improvement in quality and quantity of agricultural products, burst economy and create jobs by making farming more attractive to people. It will benefit the user in a way that provide the user with the solution instantly with even low price smart phone (android) resource without having to wait for the human experts (Agriculturist and/or Agricultural engineer). The client android phone application is design and developed for android phones using java programming language. Android studio with Android SDK tools (maximum API level 23 Support) which is java based programming language software is used in designing and implementing the application software. This mobile phone application software supports all versions of androids including Android 7.2 "KitKat". The developed application software has two major

functionalities button: Image capture from camera/image selection from gallery and upload image to server.

4.1.1 IMAGE CAPTURE FROM CAMERA/IMAGE SELECTION FROM GALLERY

After clicking the icon, the very first page of the application shows a basic layout content_main.xml having one image view feature and two buttons. One button is for snapping the image with mobile phone camera and the second one is for selection of image from the mobile phone gallery as shown in the fig 4.1 below.

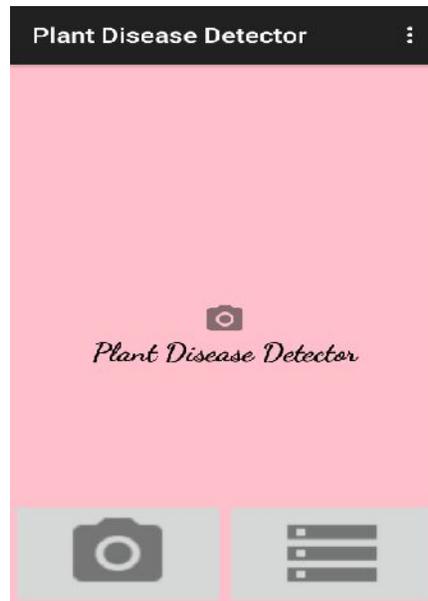


Figure 4. 1: Interface of the Application

Once the apps is started, the MainActivity.java class starts and the On Create() method sets the layout view to content_main.xml. The intent ACTION_PICK is sends automatically to get the image from external storage of the phone by this class. On the image view, the bitmap is set as shown in fig4.1.2 and fig4.1.3 below.



Figure 4. 2: Choosing the picture from gallery



Figure 4. 3: Selected image showing in image view

4.1.2 UPLOADING IMAGE TO THE SERVER

The main function of this button is to upload the image to the server for upward forwarding to the Matlab. Image uploading is done through the use of AsyncTask class. Asynchronous task class is a special class which enables the application to perform background work and transmit the result to UI threads without interrupting the UI thread or handlers. Normally, this class is designed to be assistant class around a thread or handler which should be used for very short time operations. Asynchronous programming helps the app stay functional when it does work in the background. The downside of using synchronous programming on the UI thread to upload content to server is that the app will

be abrogates until the method returns. The app will not respond to user request which might turn the user waiting. Thus the asynchronous class is the best choice for this type of work so that the user interface will remain active while the operation is done in the background. After completion of the task, the application will acknowledge that the image has been uploaded to server by showing image uploaded successfully on a toast and if there is a problem it will display error message in the toast. This asynchronous task mainly overpowers three methods that is pre Execute(), post Execute() and do In Background(). Firstly, the bitmap image is transform to string using the built in method Base64.encodeToString () method. It changes the bitmap image to Base 64 strings so that it could be transmitted to the server using HTTP web services. The bitmap image is initially compressed into byte array output stream before putting it into a byte array. Then this array is changed to base 64 strings. The image in string form is kept into a Hashmap data structure which will be transmitted as data using HTTP URL connection. The connection between the application and the server is done by the RequestHandler.java class. It establishes the connection link using Http URL Connection class in java. Upload image class transfer the Hash map and server address to RequestHandler.java class

The connection time out is set to 10 seconds and the request method type is POST. In the server side, the server developed using java programming language on netbeans then verifies if the request is POST type. The data is then decoded from string to image and sent to MATLAB for processing if the request is POST type.

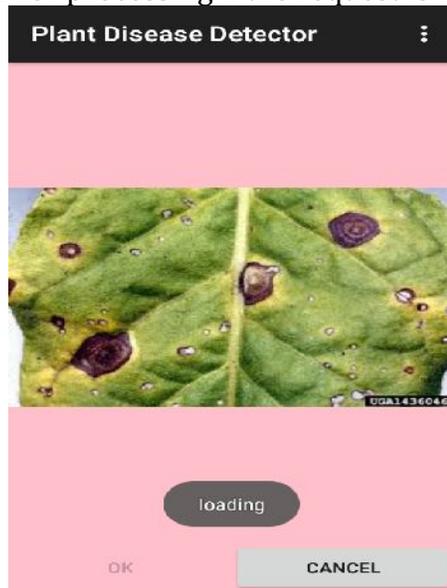
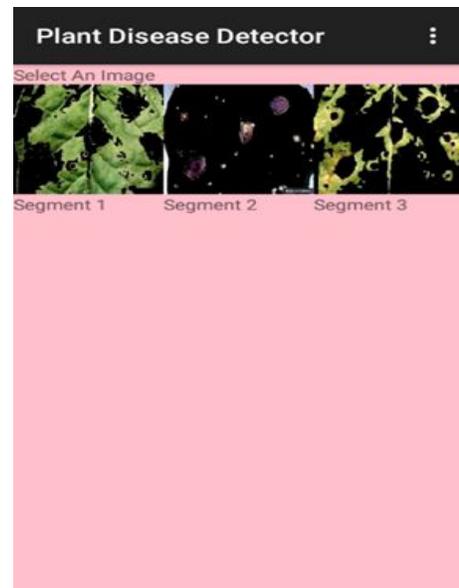


Figure 4. 4: Uploading the image to server



4.1.3 DOWNLOADING RESULT FROM THE SERVER

Once the mobile phone completed the image uploading successfully, automatically the MATLAB will sent the three segmented image (this work use k-means clustering algorithms and the value of k is 3) back to the mobile phone waiting for the user to choice one as shown in figure 4.1.5 below.



Figure 4. 5: the three segmented image displayed on the phone for the user to choice

Once the user choice any one, then the selected (choosing one) segmented image is sent to the MATLAB for processing via the server. If the MATLAB completed the processing and the classification result is ready. The server will receive the result in form of string image and text string from the MATLAB automatically and the mobile phone will receive the result from the server on its layout which is set to activity_view_image.xml and the ViewImage.java class starts executing as shown in fig4.1.6 below.

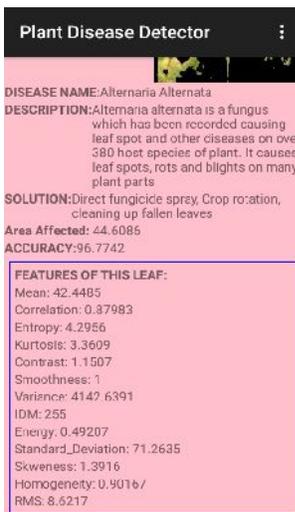


Figure 4. 6: Result of test plant leafs

Here also asynchronous programming is use for the same reason as we need the task to operate and function in background. The input stream is converted from string to bitmap format. The request method type is GET and the URL connection time out is set to 10 seconds. In the activity_view_image.xml layout have three text views and one image view, one text view for displaying the result, the second for displaying the brief explanation of the disease, the third one for displaying the solution and the one image view for displaying the segmented image. Both uploading and downloading image activities are done with the help of asynchronous programming and the RequestHandler.java class. The connection between the android application and the server is established by send Get Request() method in Request Handler.java class and the input stream is decoded inside the ViewImage.java class inside asynchronous programming.

4.1.4 CLIENT AND SERVER INTERACTION

This section of the system is executed once the client sends the images to the server via HTTP protocol using mobile phone application. The image received by the server using servlet script is stored in the assigned directory and the image processing is start on the assigned image automatically.

4.1.5 SHELL EXECUTION

For this research work, a local server was created in laptop using “TOMCAT server” software. The server has servlet scripts that will automatically receive and store image from the client (mobile phone device) into a particular directory and then send the image to MATLAB for pre-processing and segmentation. Before, an ‘exe’ file of MATLAB code was created and saved to local server. From the servlet script with function called exec(), the ‘exe’ file of MATLAB is executed through shell execution. The exec() method in the servlet is called immediately when image is receive and store in the server, so the whole process is done automatically without any human interaction on our server side. It reduces the possibility of errors in a way that we can depend on machines here completely. The MATLAB ‘exe’ file contains every step of the image processing algorithms.

4.1.6 PARSING THE RESULT

The servlet script fetches the data (the results) from the MATLAB as soon as the MATLAB code is done with the processing through shell execution and the results is ready and save in assign directory in the server. The servlet script fetches the data for the mobile application for request method type GET and sends to client. The result has two types of data. One is the segmented image which highlights the affected portion of the plant in Base 64 string form and the other part of the result is the disease name, brief description of the disease and possible solution about the the disease (all the three is contained as one in array form) in the MATLAB code generates after analyzing the image in string form. Both of them are fetched by servlet script and sent over the HTTP to the mobile application.

4.1.7 CLASSIFICATION

First all the necessary features of different plants with the various disease symptoms (**Alternaria Alternata, Anthracnose, Bacterial Blight, Cercospora Leaf Spot and Healthy Leaves**) is extracted and stored in pre-defined storage location. MAT file which is called training data sets. And then all the necessary features of plant to be tested is extracted so that the MATLAB will compare this features with the one of the training data

set to determine the one that match with the testing data set using Multiclass Support Vector Machine (Multiclass SVM) classifier for classifying the disease as **either Alternaria Alternata, Anthracnose, Bacterial Blight, Cercospora Leaf Spot or Healthy Leaves**. This Multiclass SVM use to assign labels to instances by using support vector machines, where the labels are drawn from a finite set of several elements. The dominant approach for doing so is to reduce the single multiclass problem into multiple binary classification problems. Common methods for such reduction include: building binary classifiers which distinguish between (ii) between every pair of classes (one-versus-one) or (i) one of the labels and the rest (one-versus-all).

In this case, the classification of new instances or input images for the one-versus-all case is done by a winner-takes-all strategy, in which the classifier with the highest output function assigns the class and detects the particular disease.

4.3 FINDINGS AND DISCUSSIONS

The results obtained are explained in details in this section. For this research work, a personal computer that has Windows 8, 4GB of RAM, 500GB of HDD, an Intel(R) Core(TM) i5 CPU and Android Mobile phone is used for performing the experiments. MATLAB Software, netbeans and Android Studio are used for the simulation of work and designing the interface of the Android Apps for Mobile phone. The automated System for Plant Disease Detection Using Image Processing and Machine Learning Integrated with Mobile Application starts with capturing of digital high resolution images. More than 500 different plant leaves' images with different disease symptoms (Cercospora Leaf Spot symptoms, Bacterial Blight symptoms, Anthracnose symptoms, Alternaria Alternata symptoms, and Healthy leaves) are obtained from <https://github.com/spmohanty/plantvillage-Dataset> and store in the Database for the experiment. Then the images were applied for pre-processing for image enhancement. Captured leaf images are segmented using k-means clustering method to form clusters. Features are extracted using GCCM before applying K-means and SVM algorithm for training and classification.

The main goal of carryout this research is to develop an automated System for Plant Disease Detection Using Image Processing and Machine Learning Integrated with Mobile Application that can detect and classify a given unknown plant leaf image as either Cercospora Leaf Spot, Bacterial Blight, Anthracnose, Alternaria Alternata, or Healthy leaves. The experiment is first performed by converting the RGB images into HSI, GLCM is used in extracting features of the plant leaf images before apply K-means Clustering to obtained three segment of the images. SVM Classifier is used to classify the image. The percentage accuracy varies from one plant leaves to another but with SVM Classifier; the average percentage accuracy obtained is 97.01%.

The result obtained for this research work produced high percentage accuracy compare to the percentage accuracy obtained by Sharada, David, & Marcel, (2016) that are able to achieved only average percentage accuracy of 31.40%, because the system used deep convolutional neural network (KNN) for training and classifier. The data source for both the work of Sharada et al. (2016) and this research is obtained from the same source. (Bababe et al. 2017) able to achieved 73.648% percentage accuracy which is low compare to this research that has average percentage accuracy of 97.01%. (Bababe et al., 2017) use Artificial Neural Network for classification while this research uses Multi-Class SVM.

4.5 CONCLUSION AND FUTURE WORK

Even though the work gives satisfactory output, more work need to be done to cover more plant disease. And also this work is limited to only plant leaves, more work can also be done to cover stems and roots of the plants.

REFERENCE

- Bhong, V. S., and Pawar, B. V. (2016). Study and Analysis of Cotton Leaf Disease Detection Using Image Processing. *International Journal of Advanced Research in Science, Engineering and Technology*, 50 (5), 34-45.
- Jha, A. K., Bababe, A. B., and Ashok, K. S. (2017). Plant Disease Information Generating System Based On Android Application Technology. *International Journal of Science and Research (IJSR)*, 6 (4), 2319-2464.
- Sandesh, R., and Amit, F. (2017). Plant Disease Detection in Image Processing. *International Journal of Innovative Research in Science, Engineering and Technology*, 6 (6), 2347-6710.
- Sharada, P. M., David, H., and Marcel, S. (2016). Using Deep Learning for Image-Based Plant. *Digital Epidemiology*.
- Suhaili, B. K., Noor, E. A., Hadzli, H., A'zraa, A. A., Aida, S. K., Tuan Norjihan, T. Y., et al. (2013). Classification of Watermelon Leaf Diseases Using Neural Network Analysis. *IEEE Business Engineering and Industrial Applications Colloquium (BEIAC)*, 8 (2), 459-464.
- Trimi, N. T., and Sushma, K. (2017). Plant Disease Detection Using Different. (pp. 103-106). Proceedings of the Second International Conference on Research.
- Xin Yang, T. G. (2017). Machine learning in plant disease research. *European Journal of BioMedical Research*, 3 (1), 6-9.