Detecting Jute Plant Disease Using Image Processing and Machine Learning

Zarreen Naowal Reza[1], Faiza Nuzhat[2], Nuzhat Ashraf Mahsa[3], Md. Haider Ali[4]
Department of Computer Science and Engineering
BRAC University
Dhaka, Bangladesh
auroraprova@gmail.com[1], faiza.joyee@gmail.com[2], nuzhat.mahsa@gmail.com[3], haider@bracu.ac.bd[4]

***Abstract*— Detecting stem diseases of plants by image analysis are still in an inchoate state in the research field. This research has been conducted on detecting the stem diseases of jute plants which is one of the most important cash crops in some of the Asian countries. An automated system based on an Android application has been implemented to take pictures of the disease affected stems of jute plants and send them to the dedicated server for assaying. On the server side, the affected portion from the image will be segmented using customized thresholding formula based on hue-based segmentation. The consequential feature values will be extracted from the segmented portion for texture analysis using color co-occurrence methodology. The extracted values will be compared with the sample values stored in the pre-defined database which will lead the disease to be identified and classified using Multi-SVM classifier. At the final step, the classification result along with the necessary control measures will be sent back to the farmer within three seconds through the application on their phone.**

Keywords—hue based segmentation, color co-occurrence methodology, texture analysis, Multi-SVM classifier, Android application

I. Introduction

Plant disease detection using image processing techniques has been playing a role of immense importance in agricultural industries for the recent years. Jute is considered as the Golden Fiber of Bangladesh for its huge contribution to our economy in past though the production of jute has been hurdled by various factors from the recent years. One of the factors is the inaccessibility of expert's opinion to the root level cultivators or farmers. This system will help the farmers to get accurate solution in order to cultivate healthy crops. Although visual supervision by experts has been considered as the primary way of detecting crop diseases, the use of technology can alleviate the level of reliability as well as save valuable crop of the diligent cultivators by providing correct information in the fastest way.

This research will be beneficial to the farmers who struggle to get expert's guidance and suggestions to keep their crop safe from several fatal diseases. The mobile application interface of this system is very user friendly and people with no prior technological knowledge can use this system and get benefited by it. In this research, the five most common diseases have been detected followed by providing respective control measures. The diseases are named as Anthracnose (Colletotrichum Corchori), Black band (Botryodiplodia Theobromae), Die back (Glomerella Cingulate), Stem rot (Macrophomina Phaseolina) and Soft rot (Sclerotium Rolfsii). By using various image processing techniques, the system can recognize these diseases from the image of the affected stem and provide users with disease identification and proper control measures within three seconds. Therefore, the system has every scope to serve the emergent field of plant disease detection by providing the classification of stem diseases within a short period of time with higher level of accuracy and a very simple user interface.

The rest of this paper is arranged as follows. Section II discusses related work. Section III explains the entire system overview. In Section IV the system implementation is discussed step by step. Section V demonstrates experimental results and analysis for detecting jute diseases. In section VI outcome of a survey conducted on farmers is presented and discussed. Section VII concludes this paper with a brief summary.

II. Literature Review

Though ample number of researches has been done on detecting plant diseases using image processing and computer-vision most of them have failed to detect the stem diseases efficiently. As example, S. Arivazhagan and his co-authors have discussed about leaf disease detection using image processing and neural network where they used the Color Co-occurrence Methodology for extracting the features for texture analysis [1]. For disease classification purpose, they have used multi-class SVM (Support Vector Machine) classifier with a winner-takes-all strategy where the classifier with the highest output function assigns the class [1]. Besides, referring to the integration with mobile applications Amos Gichamba and Ismail Ateya Lukandu have described different employments of mobile applications for agricultural purpose and presented a general model for designing such systems [2]. However, their focus of interest was detecting the leaf diseases of plants rather than the stem diseases. Another mobile-application based system named ‘Beetles’ has been presented by Rahat Yasir and Nova Ahmed which has been designed to support farmers in rural area to detect crop diseases from the image captured by a cell phone in real-time using histogram and color information of the image [3]. Having said that, their

978-1-5090-2906-8/16/$31.00 ©2016 IEEE iCEEiCT 2016

Authorized licensed use limited to: University of Roehampton. Downloaded on March 17,2024 at 15:27:56 UTC from IEEE Xplore. Restrictions apply.

system detects diseases based on color compromising the accuracy of results in case of diseases with same or almost same color. On the other hand, H. Al-Hiary and his co-authors have come up with the detection process of stem disease in such way that their system requires the background of the picture to be plain or unicolor [4]. Regrettably, as their segmentation method is based on masking the green pixels the efficiency has been compromised for most of the cases where stems turn brownish in color at matured stage [4]. To overcome this defect Y. Sanjana and her co-authors have proposed a system to detect different leaf diseases of crops using image analysis consisting of mathematical morphology for segmentation, texture, shape and color feature extraction for classification of diseases [5]. Nevertheless, their system needs the intervention of respective experts to give review and mention the level of severity by examining the extracted features which can increase the period of time needed for getting the ultimate result [5].

III. SYSTEM OVERVIEW

This system is designed in such way that the client does not require much technical knowledge to operate it. The instructions are pretty straight forward and will lead the user to the desired solution within short time. This automated system is mostly combined of press and send techniques.

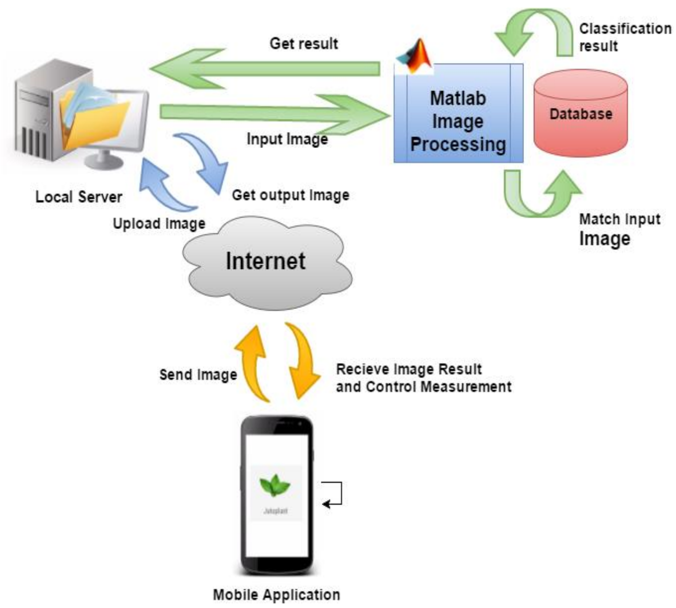


Fig. 1. System Overview

Farmers or users will take image using mobile phones and send it to server via mobile application. In server, the image will be analyzed based on visual and textural attributes using different image processing algorithms and the system will yield the disease name which will be sent to the mobile application on the client's end.

IV. SYSTEM IMPLEMENTATION

A. Mobile Application Developement

Android studio with Android SDK tools (maximum level API 23 support) was used in creating and implementing the functionalities, this application supports all versions of Android operating system. This mobile application has four major functionalities explained below-

1) *Image capture/selection:* At first, after clicking the icon, the application shows a basic layout having one image view feature and two buttons. One button is for uploading the image to the server and the second one is for downloading the result from the server. At the very beginning, an intent ACTION_PICK is sent to get the image from external storage of the phone. Then that bitmap image is set on the image view.

2) *Upload image to the server:* The upload image button for uploading the image is pressed in order to upload the image to server. Image uploading is done with the help of AsynchTask class. Asynchronous task class is a special class which allows the application to perform background work and publish the result to UI threads without interrupting the UI thread or handlers. Asynchronous programming helps the app stay responsive when it does work in the background. Firstly the bitmap image is converted to string using the built-in method method. The bitmap image is firstly compressed into byte array output stream and then put into a byte array. Then this array is converted into base 64 string format. The image in string form is put into a Hashmap data structure which will be sent as data using Http URL connection. The connection between the application and the server is maintained by using HttpUrlConnection class in java.

3) *Download the result from the server:* Once the image processing is completed and the disease is detected. The server sends back the result in form of string image and text string. After uploading is completed, the Download the result button is pressed and the layout is set to new page. Here also we use asynchronous programming for the same reason as we need the task to operate in background. The input stream is decoded from string to bitmap format . The next layout contains two buttons, one image view and one text view . The top button is for getting the segmented image which shows only the affected area of the plant and the bottom button is for navigating the user to solution needed for the disease. The image view shows the segmented image and the text view updates with the name of the disease the plant is diagnosed with.

4) *Getting solution:* After clicking the button, the user is directed to solution.xml layout where he or she can see the control measurement recommended by the specialist in Bangladesh Jute Research Institute (BJRI). The instructions are simple enough to understand and the occurrence period of the diseases is also included

B. Client and server interaction

The system will be executed in the server as soon as the client sends the image via mobile application. When the

Authorized licensed use limited to: University of Roehampton. Downloaded on March 17,2024 at 15:27:56 UTC from IEEE Xplore. Restrictions apply.

mobile application sends data to the server, it is received using a PHP script using methods and the fetched images are stored in the assigned directory and our image processing will start on the assigned image automatically.

C. *Shell execution*

'.exe' file of the MATLAB code has been created and saved in the local server. From the PHP script with a function called exec (), this '.exe' file of MATLAB is run through shell execution. The exec () method in the PHP is called just after receiving and saving the image in the server.

D. *Parsing the result*

Once the MATLAB code is executed through shell execution and the result is saved in the pre-assigned directory in our server, the PHP script fetches the data for the mobile application for request method type GET and sends it to our client. The result consists of 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 the MATLAB code generates after analyzing the image in string form. Both of them are fetched by PHP script and sent over the internet to our mobile application.

E. *Server Side Operation*

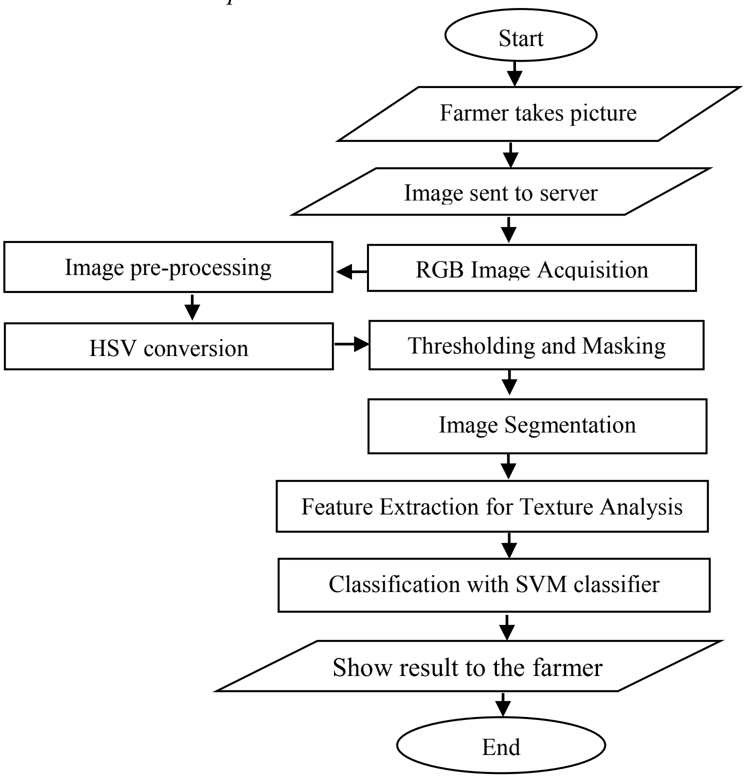


Fig. 2. Flowchart of overall process

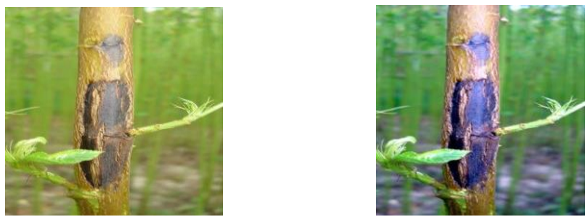
The image received by the Android application has been sent to the system server where the image would be gone through several levels of analysis.

The stages from segmenting the image to identifying the disease are described as followed for Anthracnose.

1) Image Preprocessing: Before proceeding towards image analysis the image should be processed in order to acquiring better result. Images taken from camera phones contains different factors which alters the result of the analysis. The image preprocessing procedure is performed by following certain steps consists of image resizing, image enhancement and noise removal.

a) Resize Image: To perform the classification, the size of the input image must match the size of the images stored in the database. Thus, the input image must be resized to a fixed dimension at the very first stage.

b) Enhance Image: In this step the image intensity values or colormap has been enhanced by adjusting the contrast of the image by defining the upper and lower limits of the pixel values that will be used for stretching the image. The limits are specified by considering the bottom 1% and the top 1% of all pixel values of the image [6].



a) b)

Fig. 3. A jute stem infected by anthracnose a) Original Image b) Enhanced Image

c) Noise Removal: The uploaded images may contain noise. Nosie can alter the features of an image and leads to obtain unexpected and deceptive result. Thus it is very important to remove the noises from the image. In this research, a bilateral smoothing filter has been used for noise cancellation. The bilateral filter is the technique which smoothens the image by replacing the intensity value of each pixel with the weighted average of the intensity values of neighboring pixels [7].

2) Hue-based segmentation: In this paper, the hue-based segmentation method has been used to segment only the affected portion from the image. As the entire process has been conducted on stem diseases it is not possible to simply mask the green pixels from the image likewise the ones done in case of detecting leaf diseases. Therefore, the hue-based segmentation method has been applied along with a customized thresholding formula.

a) HSV Conversion: At first, the RGB spaced image has been converted to HSV color space. In HSV, hue image conveniently represents the original color whereas saturation works well to mask the image and extract the region of interest [8]. Then the individual channels have been extracted to separate the hue, saturation and intensity images.

Authorized licensed use limited to: University of Roehampton. Downloaded on March 17,2024 at 15:27:56 UTC from IEEE Xplore. Restrictions apply.

a) b) c)

Fig. 4. a) Hue, b) Saturation and c) Intensity Image

b) Thresholding: Image thresholding is an effective way of dividing an image into a foreground and background. In this research, a threshold formula has been introduced to mask only the disease affected part of the stem from the image. At first, the saturation image has been manipulated with a theory described by R.C Gonzalez where a binary mask is generated by thresholding the saturation image with a threshold value which is equal to ten percent of the maximum value of the image [8]. Any pixel value greater than the threshold value is set to 1 (white) and all others are set to 0 (black) [8]. After that the masked saturation image has been multiplied with the hue image. Now at this point the region of the stem from the image should be separated. However, this approach solely does not accomplish the goal to extract only the disease affected portion of the stem and discard the rest. To achieve the goal, the product image has been masked again with a threshold value of 0.5 as the pixels with value greater than 0.5 prominently identify the disease affected region. Therefore, the comprehensive formula for thresholding appears as followed**.**

MaskedImage = M > 0.5

where M = Hue * (Saturation >(Max(Saturation) * 10%))

Fig. 5. Masked Image

c) Blob Detection: The masked image still contains some unwanted regions which are definitely needed to be get rid of. For that purpose, the morphological analysis has been performed which includes erosion and dilation and for the largest connected component has been extracted using eight-way local neighborhood measurement.

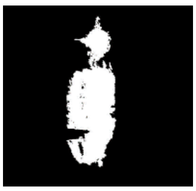


Fig. 6. Segmented Blob

d) RGB Conversion: For further analysis, the segmented portion needs to get back to its original color by being converted to RGB color space.

Fig. 7. Segmented RGB Image

3) Feature Extraction: The significant features from the image need to be extracted in order to perform the texture analysis. For that purpose, the color co-occurrence methodology has been used which is developed through the GLCM (Grey-level Co-occurrence Matrices). GLCM is used for sampling an image statistically in a way certain grey-levels occur in relation to other grey-levels [1]. It also provides feature information about the position of the pixels of the image in relation to their neighboring pixels. For this research, thirteen [9] feature values have been reckoned for each of the training images and the input image to perform texture analysis. The features are mentioned below:

1. Contrast
2. Correlation
3. Energy
4. Homogeneity
5. Mean
6. Standard Deviation
7. Entropy
8. RMS (root mean square) contrast
9. Variance
10. Smoothness
11. Kurtosis
12. Skewness
13. IDM (Image difference-measure)

The above features are calculated from the GLCM using their corresponding formulas.

4) Classification: After the extraction, the necessary features are compared with the pre-calculated dataset stored in a .mat file. The SVM (Support Vector Machine) classifier is used for classifying the disease. Support vector machines (SVMs) are a set of related supervised learning methods used for classification and regression [1]. Supervised learning refers to the process where a given set of labeled observations are stored in the database and an unknown sample has been analyzed and compared to assign it with a label in accordance. As there are five diseases to be dealt with there must have to be five classes. Thus a multi-class SVM classifier has been used. In this case, the classification of new instances or input images has been performed where the classifier with the highest output function assigns the class and detects the particular disease.

5) Create Database and Train Classifier: In this research, two .mat files have been generated, namely: (i) Training data, and (ii) Result.

Authorized licensed use limited to: University of Roehampton. Downloaded on March 17,2024 at 15:27:56 UTC from IEEE Xplore. Restrictions apply.

The dimension the of training dataset can be defined as 150x13 where 150 is the number of rows and 13 is the number of columns. The 13 columns contain thirteen feature values of each of the training images. Similarly, result file contains 1 row with 150 columns each holding a digit from 1 to 5 representing the five diseases - Anthracnose, Black band, Die back, Soft rot and Stem rot respectively.

After that, a software program written in MATLAB loads .mat files representing the datasets, train the classifier and then use it to execute the classification task on the test data.

V. EXPERIMENTAL RESULTS AND OBSERVATION

For our research, we have examined with two different cases for extracting the features for texture analysis from the training images. In case 1, we have calculated the features from the training images without applying the hue-based segmentation method on them. In this case the diseased portion is not segmented from the image and the features are calculated along with the background, unwanted portions and noise of the images. As a result, when the segmented part of the test images is classified based on those feature values, it provides a dissatisfactory classification outcome. On the other hand, in case 2 at first the diseased portion is segmented from the training images after being preprocessed and then the features are extracted. The latter case succeeds in providing a satisfactory outcome in classifying the diseases.

From the observation of both cases, we found that case 1 provides us with 60 percent of accuracies in detecting the diseases whereas case 2 provides an accuracy of around 86 percent which is far better than case 1. Comparing both the accuracy results, we have decided to proceed with the method applied in case 2.

The comparison diagram between two cases based on detecting five diseases is shown in Fig.8. In the figure, for case 1 the percentage of detecting Anthracnose as Anthracnose is 62.50%, detecting Black Band as Black Band is 33.33%, detecting Die Back as Die Back is 60%, detecting Stem Rot as Stem Rot is 98% and detecting Soft Rot as Soft Rot is 50%. On the contrary, for case 2 the percentage of detecting Anthracnose as Anthracnose is 87.90%, detecting Black Band as Black Band is 66.66%, detecting Die Back as Die Back is 80%, detecting Stem Rot as Stem Rot is 100% and detecting Soft Rot as Soft Rot is 87.50%. Fig.9 displays the total amount of accuracy in percent based on detecting all the diseases as they are expected to be. In the figure, case I provides an accuracy level of 60% whereas case 2 performs with an accuracy of 86%.

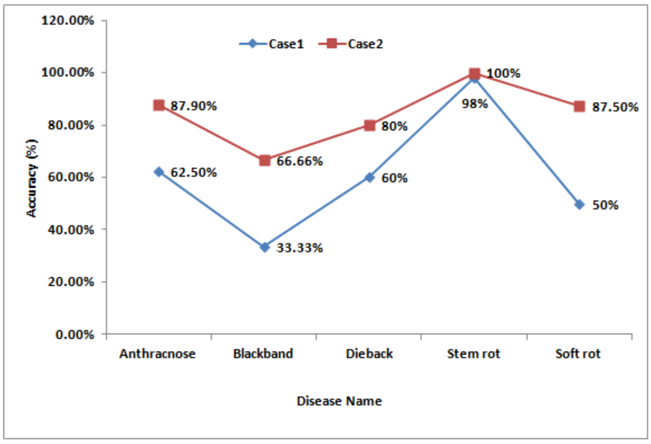


Fig. 8. True positive comparison between two cases

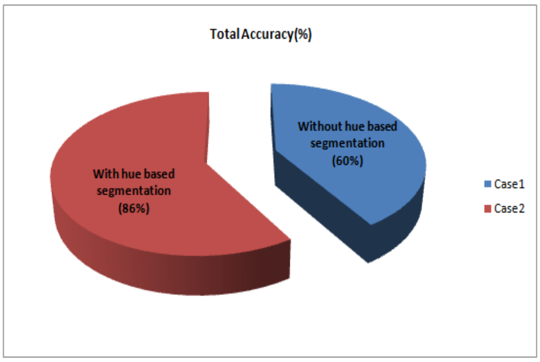


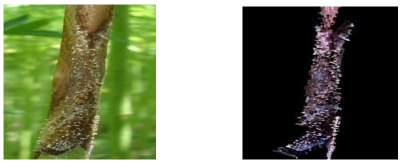
Fig. 9. Ture positive comparison between two cases based on total outcome

In both cases the following formula is used

Total percentage of accuracy = (C x 100) / N

Where, C = no. of correct outcome

N = no. of test image



a) Soft Rot

b) Black Band

c) Die Back

Authorized licensed use limited to: University of Roehampton. Downloaded on March 17,2024 at 15:27:56 UTC from IEEE Xplore. Restrictions apply.

d) Stem Rot

Fig. 10. the original images of a) soft rot, b) black band, c) die back and d) stem rot along with their segmented image

a b c

Fig. 11. a) select image from gallery b) upload image c) aqcuire result

VI. EVALUATION

We have conducted a short evaluation test on five farmers from BJRI (Bangladesh Jute Research Institute) where we asked them to use the application and rank it out of 5 based on the following criteria. The outcome is as follows.

TABLE I. FARMER FEEDBACK

User no.	Evaluation Criteria			
	User-friendly (0-5)	***Preference over/along with manual inspection (0-5)***	***Bug/Error (0-5)***	***No. of correct outcome (5 test images)***
1	3	2	4	4
2	4	4	4	4
3	3	3	3	3
4	2	2	4	4
5	4	4	5	5
Avg	**3.2**	**3.0**	**4.0**	**4.0**

The average outcome gained from the evaluation test provides an overall impression about the efficacy of the system where three out of five farmers preferred using this application to detect diseases along with manual inspection. Most importantly, the accuracy level obtained from the hands-on use (80%) almost matches the accuracy level measured from the test images (86%).

VII. CONCLUSION

In this paper, we have built an automated system to deduce stem oriented diseases for jute plants using image segmentation and feature extraction with along with potential machine learning. Application of machine learning specially image analysis and texture analysis in practical cases are now more common and encouraged than ever before. Although visual analysis done by human is simpler technique but it cannot be accessible always. On the other hand, providing the farmers with an automated and reliable system for crop disease detection to be used from their mobile phone can bring an insurgency for agricultural industry. The research work does not contain disease detection technique for the leaves. In future, it will be a great challenge for us to build a universal app that can be used to detect any sort of disease of the jute plants considering both stems and leaves.

ACKNOWLEDGMENT

We want to thank Dr. Md. Mahbubul Islam, Chief Scientific Officer and Dr. Saleh Mohammad Ashraful Haque, Senior Scientific Officer, Bangladesh Jute Research Institute (BJRI) for providing us with necessary dataset and suggestions.

REFERENCES

[1] S. Arivazhagan, R. Newlin Shebiah, S. Ananthi, and S. Vishnu Varthini, “Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture features,” Agric Eng Int: CIGR Journal, vol. 15, no. 1, pp. 211–217, Feb. 2013. [Online]. Available: http://jp.mathworks.com/matlabcentral/answers/uploaded_files/25743/FinalAlgorithm.pdf. Accessed: Mar. 27, 2016.

[2] A. Gichamba, I. A. Lukandu “A Model for designing M-Agriculture Applications for Dairy Farming”, The African Journal of Information Systems, Volume 4, Issue 4, 2012.

[3] R. Yasir and N. Ahmed, “Beetles: A Mobile Application to Detect Crop Disease for Farmers in Rural Area,” presented at the Workshop on Human And Technology (WHAT), Khulna, Bangladesh, Mar. 8, 2014, Academia, pp. 11–14. [Online]. Available: http://www.academia.edu/6370973/Beetles_A_Mobile_Application_to_Detect_Crop_Disease_for_Farmers_in_Rural_Area. Accessed: Mar. 21, 2016.

[4] H. Al-Hiary, S. Bani Ahmad, M. Reyalat, M. Braik, and Z. AlRahamneh, “Fast And Accurate Detection And Classification Of Plant Diseases,” International Journal Of Computer Applications (0975-8887), vol. 17, no. 1, pp. 31–38, Mar. 2011.

[5] Y. Sanjana, AshwathSivasamy, and SriJayanth, “Plant Disease Detection Using Image Processing Techniques,” IJIRSET, vol. 4, no. 6, pp. 295–301, May 2015.

[6] Available at: http://www.mathworks.com/ Accessed: Mar. 2, 2016

[7] S. Paris, P. Kornprobst, J. Tumblin, and F. Durand, “A Gentle Introduction to Bilateral Filtering and its Applications,”. Available at: https://people.csail.mit.edu/sparis/bf_course/course_notes.pdf Accessed: Mar. 19, 2016

[8] R. C. Gonzalez, R. E. Woods, D. J. Czitrom, and S. Armitage, Digital image processing, 3rd ed. United States: Prentice Hall, 2007.

[9] “Plant disease classification”Available at: http://www.mathworks.com/matlabcentral/fileexchange/50624-plant-disease-classification/ Accessed: Mar. 5, 2016

Authorized licensed use limited to: University of Roehampton. Downloaded on March 17,2024 at 15:27:56 UTC from IEEE Xplore. Restrictions apply.