International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys.13, No.1 (May-23)

http://dx.doi.org/10.12785/ijcds/1301104

A Deep Learning Based Smartphone Application for Detecting Mango Diseases and Pesticide Suggestions

Md. Naimur Rahaman[1], Musthafezur Rahaman Chowdhury[1], Md. Ashiqur Rahman[1], Humayra Ahmed[1], Md. Mamun Hossain[1], Md. Hasibur Rahman[1], Md. Shovon Biswas[1], Md. Abdul Kader[1], Tanvir Ahmad Noyan[1] and Milon Biswas[1]

[1]*Department of Computer Science & Engineering, Bangladesh University of Business & Technology, Dhaka, Bangladesh*

Received 7 Jun. 2022, Revised 31 Mar. 2023, Accepted 23 Apr. 2023, Published 30 May. 2023

Abstract: Mango trees are tropical and subtropical trees that flourish in warm climates. It is a popular, tasty fruit as well as a cash crop. Farmers have a hard time selling their products when their output is reduced owing to diseases that affect mango trees. To improve quality and production, it's vital to address any harmful illnesses as soon as possible. This problem prompted the development of novel technologies for detecting and diagnosing mango plant diseases, as well as expert systems for disease prevention. Three machine learning techniques are employed to detect mango diseases in this paper. A dataset with 20 different classes of infected and healthy mango fruit and leaf photos has been created. Among these machine Learning methods, DenseNet169 obtains the highest accuracy of 97.81%, with precision, recall, and F1-scores of 97%, 96%, and 96%, respectively. An Android app has been developed and coupled with the machine learning model that aids in the identification of mango illness as well as the recommendation of pesticides based on disease detection.

Keywords: Image Processing, Data Augmentation, Machine Learning, DenseNet169, InceptionV3, MobileNetV2

1. INTRODUCTION

Agriculture has a significant role in economic growth in today's globe. One of them is the mango (Mangifera indica). Mangoes are mostly grown in warm tropical and subtropical areas in Asia, Latin America, and Africa. India (40.28 %), China (10.23 %), Thailand (7.36 %), Mexico (4.71 %), Indonesia (4.68 %), Pakistan (3.44 %), Brazil (3.04 %), Egypt (2.73 %), Bangladesh (2.49 %), and Nigeria (1.96 %) have dominated the export market [1] . This fruit is the world's fifth-most cultivated fruit. However, every year, a large volume of mango is ruined by various insects and illnesses. One of the key causes of this massive loss is mango sickness. Farmers lose mango fruits every year owing to a variety of viral, bacterial, and fungal diseases. Finding a cure for these diseases as quickly as possible is one of the most difficult challenges[2].

Mango fruit production is now dropping as a result of climatic conditions and environmental impediments such as heavy rainfall, high humidity, soil nutrient depletion, and a number of related illnesses and disorder difficulties. Several diseases affect the mango plant, such as gall infestation, Webber's attack, mango malformation, stem miner, anthracnose, Alternaria leaf spots, etc. Among these diseases, bacteria are the most common cause, followed by viruses, fungi, parasites, and even unfavorable environmental conditions. When a disease attacks a leaf, it disrupts the photosynthesis process, resulting in plant death. Mango trees began to wilt all around in the early stages of the illness, and twigs started to die back. When the illness is advanced, it manifests as the curling and drying of leaves, full defoliation of the tree, and staining of vascular tissues along stems and branches. In order to lessen the devastation caused by the dieback disease on mango trees, the first step was to identify the fungus based on its morphological and cultural traits. [3].

The low fruit production of mango fruits is due to the inability of illiterate farmers to identify diseases affecting mango plants[4]. It can be difficult for farmers in rural areas to reach experts if they need to travel a substantial distance. Many farmers are unaware of many diseases, making this process too expensive and time-consuming. Common natural causes of plant disease include insects, varying weather, temperature, etc. As a result, the disease must be detected as early as possible for treatment and prevention. It is common for disease symptoms to be visible on the stems, fruits, and leaves of plants. Now that artificial intelligence has made it possible to detect disease in plants from raw images, the disease can be detected by the human eye. The goal of this paper is to give a visual representation of the condition so that current information on the disease and how to prevent

E-mail address: naimurrahaman248@gmail.com, musthafizchowdhury@gmail.com, ashiqashiqur@gmail.com, humayraahmed833@gmail.com, mamunhasan.cs@gmail.com, hasiburrahman875@gmail.com, shovonbiswas.cse@gmail.com,abdulkadirshohan@gmail.com, tanvirnoyon.bubt@gmail.com, milon@ieee.org

it may be found more easily. We utilize the data we collect in our research.

The kind of disease is determined by the symptoms and the afflicted leaf area. Previously, plant diseases were often identified by constant monitoring of plants by farming specialists. In the case of small farms, diseases might be quickly identified and rapid preventative and control measures implemented. However, in the case of big farms, it is time-consuming and costly. As a result, finding an autonomous, accurate, rapid, and less expensive technique for plant disease diagnosis is critical. Deep learning employing Neural Networks is a subset of machine learning. It has expanded its wings in a variety of disciplines, offering a wide range of applications. In this work, we created an application that can identify diseases and recommend pesticides. As computer technology develops, it can help farmers monitor and manage plant diseases. The overall contribution of the paper includes:

- Building a dataset that includes both mango leaf and fruit.
- Applying and Validating different machine learning models on our dataset.
- Developing a mobile application that can detect mango diseases using captured images and suggest pesticides according to disease.

The rest of the paper is segmented as follows. Section 2 discusses about related works. Section 3 provides a detailed description of the proposed system. Section 4 discusses about the evaluation metrices to evaluate our system. Section 5presents the performance evaluation of the proposed architecture.Section 6 discusses about the implementation of our Android Application. Finally, Section 7 concludes the paper.

2. RELATED WORKS

Arya et al. [2] proposed a disease detection method for potato and mango leaves. They applied different deep learning and machine learning methods such as AlexNet, GoogleNet, DenseNet, etc in their experiment. They achieved about 98.33% accuracy with AlexNet but this requires a more practical solution. Ashok et al.[3] presented a pre-trained deep convolutional neural network (CNN) architecture. In their experiment, the maximum training accuracy was 98.6% and validation accuracy was 96.4 %. They had achieved good accuracy but their computational and training time is slow. Multi-SVM and GLCM techniques were suggested by Tumang [4]. By recognizing pests and diseases through leaf and fruit markings, they helped mango farmers in Pampanga by improving crop management, particularly in the application of pesticides, and addressing one of the key causes of the significant drop in the Philippines' mango production. They made use of their own data. They measured an 85 percent accuracy rate for pest identification using multi-SVM. S.Veling et al.[5] proposed a method that enables the identification and recognition of mango tree diseases, as well as the provision of appropriate preventative care and treatments. They applied the Support vector machine method, and they achieved an accuracy of 90% but the sample size is minimal. Wongsila et al. [6] proposed an approach for developing and designing an algorithm for detecting anthracnose-infected mangoes. Using CNN, they were able to achieve a 70% accuracy. The accuracy was not good enough, and so was the dataset. Jos et al. [7] presented a method to detect diseases in mango trees. They used K-means clustering to evaluate their experiment. They attained a 90% accuracy rate. They may increase the region of interest's healthy portion's accuracy and adjust the shadow in the images of detection of disease in mango trees using color features to account for different lighting conditions. Abdullah et al. [8] illustrated SVM-based image processing which was used to demonstrate a strategy for recognizing mango leaf disease. They got an accuracy percentage of 68.89%. The accuracy was low because the dataset was insufficient. Rahman et al. [9] proposed a neural network ensemble (NNE) for mango leaf disease recognition (MLDR) that intended a standard approach and helped to identify diseases quickly and accurately. They employed ANN, CNN, SVM, and NNE to reach an average accuracy of 80%. To train the models, they simply employed a few samples.

Pham et al. [10] Proposed a Mango Leaves Early Disease Classification method using Hybrid Metaheuristic Feature Selection with Feed-Forward Neural Networks. They were able to get a high degree of accuracy of 89.41% using ANN. Chouhan et al. [11] proposed a method for segmenting Mango Tree Leaf Anthracnose. In their approaches, they employed Radial Basis Function Neural Network (RBFNN). In their trial, they discovered that segmentation of anthracnose (fungal) sickness was accurate, with an average specificity of 91.15 % and Sensitivity of 90.86%. Rajbongshi et al.[12] employed convolutional neural network models to diagnose mango leaf disease using a transfer learning method. They ought to employ methods like CNN and others to concentrate on more ailments in order to increase accuracy. The classification of mango leaves affected by the fungal disease anthracnose using deep learning was suggested by Ashtekar et al.[13]. The classification accuracy achieved by the CNN method was 96.16%. They should create a model that can be utilized in IoT applications such as the JeVois camera, which may be employed to monitor Mango leaves.

Arivazhagan et al. [14] suggested an Identification of mango leaf diseases using CNN. The model was trained using CNN. The recognition accuracy of the suggested CNN model is 96.67 %. Chaugule et al.[15] built an IOT-based system that uses historical weather data and agricultural production to predict disease attacks on mango fruit crops. They used Random Forest Regression model in their method.Lagarteja et al.[16] proposed Convolutional Neural Networks(CNN) for Carabao Mango Pest Identification in the Philippines. CNN was used to train the model. When it came to identifying mango pests, the model had a stunning accuracy rate of 88.75%. The model was trained using a sample of 5100 images. They could add more data. They also could be trained with a different model.

Sutrodhor et al. [17] suggested a Mango Leaf Ailment Detection (MLAD) system based on a Neural Network Ensemble (NNE) and SVM. They had made use of their own dataset. They achieved an accuracy of 80%. They might improve identification accuracy by adding Comparing the texture's basic characteristics and Color features. Türkolu et al. [18] suggested a deep learning-based feature for detecting plant diseases and pests. They demonstrated and compared two techniques for vision-based automated pest detection and identification utilizing distinct learning methodologies. SVM, ELM, and KNN algorithms were utilized, with SVM classification achieving the best degree of accuracy with a resnet50 model accuracy of 97.86 percent. To improve accuracy, other sorts of illnesses should be added.

3. PROPOSED METHODOLOGY

We obtained our dataset from several sources in the first step. After assembling the dataset, we normalized it by reducing noise from images, as we mentioned in Chapter 3.4.2. The image data was then augmented in various ways, and the text or pesticide data was organized in a sensible manner. The data is then represented using various plots and charts. Then, to improve accuracy, we used deep learning approaches. On our dataset, we used Desnet169, InceptionV3, MobileNetV2, AlexNet, and VGG16. The model is then tested and confirmed to ensure that it is correct. Finally, the optimal model for development is picked. Following that, we created our application. As part of our app development, we first developed the application's backend, and then we constructed the app altogether. The Figure 1 shows the overall workflow of our system. The Algorithm 1 presents the pseudo-code for detecting mango disease and pesticide suggestion.

Algorithm 1: Mango Disease Detection and Pesticide Suggestion

1. Take or capture mango images from mobile
2. Upload the images into mobile application
3. Image enhancement is done using noise removal, blur, finding regions of interest and blurring the background of the image
4. Detect Diseases from the image with a pre-trained model(DenseNet 169)
5. **if** $(diseasefound == true)$ **then**
    - Identify the disease and find the class id of the image Search through the cure data(in JSON format)
    - **if** $(classid == keyofJSONdata)$ **then**
        - Send the class name and pesticide data back to the user.
    - **else**
        - Send The result doesn't match our dataset.
6. **else**
    - The result doesn't match our dataset.
7. Show the output in the application.

A. Dataset Collection

In our research, we have managed about 430 different sorts of diseased and healthy photos of the mango tree, as well as their insecticides. For our datasets, we constructed 20 classes, 11 of which include photographs of infected mango leaves and 7 of which have images of infected mango fruit. The other two classes feature images of healthy mangoes and healthy fruits, respectively. We gathered our dataset from several internet sites like Krishi Batayon, Flickr[19][20] , Mendeley, Google, and other online sites are examples. Pesticide recommendations for these ailments have also been collected from various internet sites. The recommendation for the pesticides was collected from renowned websites available online. The "Krishi Batayon" is a government website working on agriculture in Bangladesh. On their website, they have provided symptoms and their pesticide. We have followed their suggestions to create the data set. We have also followed the "Plantix" website which is also an agriculture-based website where they have provided information about the symptoms and pesticides but in a manual way.

TABLE I. Number of Images in different classes

Diseases Name	No of Images
Anthracnose Fruit	2332
Anthracnose Leaf	2900
Bacterial Canker Fruit	2376
Bacterial Canker Leaf	2450
Black Rot Disease Fruit	679
Fruit Borer	1440
Gall Midge Leaf	1045
Giant Mealybug Leaf	1222
Coating Mite Leaf	1674
Cutting Weevil Leaf	2116
Mealybug Fruit	819
Mealybug Leaf	1710
Mysterious Cracking Disease Fruit	1424
Normal Fruit	1760
Normal Leaf	1795
Powdery Mildew Leaf	1290
Red Rust Leaf	1870
Scab Fruit	1092
Scab Leaf	498
Shoot Mold Leaf	1312

B. Dataset Preprocessing

The data preprocessing has occurred in three stages, dataset visualization, data noise removal and, data augmentation. These two techniques are described below.

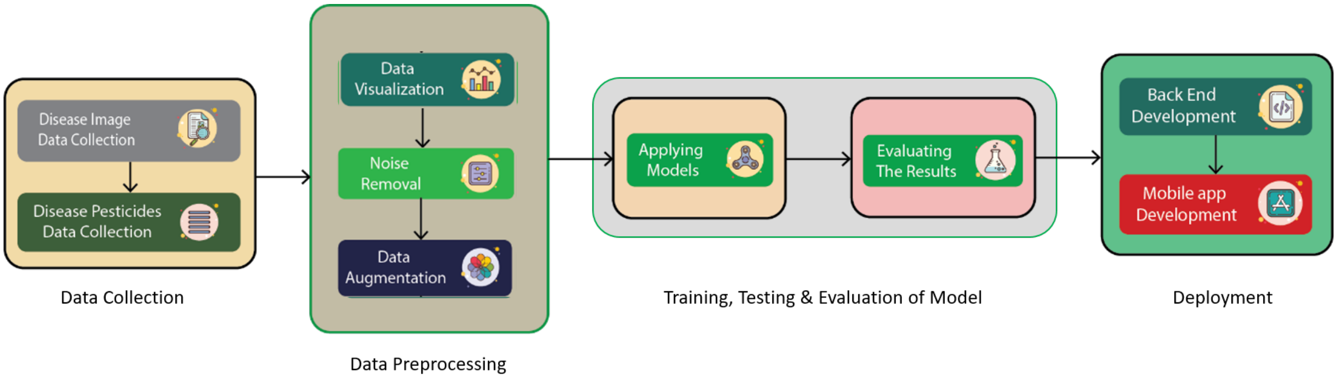

Figure 1. Workflow of the proposed system

1) Dataset Visualization

In Table I all the classes and each class's number of images are displayed. Figure 2 describes some of the photos from our data collection.

From Table I, we have seen that there is a class imbalance issue arises here for the disease name "Black Rot Disease Fruit", "Mealybug Fruit" and "Scab Fruit" as the number of collected images is few. To deal with this, we have used oversampling method. Datasets are balanced using the oversampling method by enlarging the size of rare samples. Repetition, bootstrapping, or SMOTE (Synthetic Minority Over-Sampling Technique) are used to create new rare samples rather than eliminating classes with these rare samples. We have shown the adjusted number of samples in Table II, after applying the oversampling method.

TABLE II. Number of Images updated after oversampling

Diseases Name	No of Images
Black Rot Disease Fruit	1300
Mealybug Fruit	1660
Scab Leaf	1390

2) Data noise removal

Using noise reduction as a tool for visualizing and preprocessing multidimensional images have proven indispensable in the bioimaging field, and with electron tomography in particular. Reduced noise levels are especially significant for cryptograms that suffer from low contrast and high noise levels. This includes stuck pixels in the camera or unspecific stains. Our work was helped by the [20] sites that blurred the background and removed other objects so that we could focus on the main subject. Noise is removed by focusing on the main or needed object, which results in high accuracy in detection and output.

3) Data Augmentation

Data augmentation can improve the performance and outcomes of machine learning models by producing new and diverse examples to train datasets. This would be accomplished by using domain-specific strategies to build

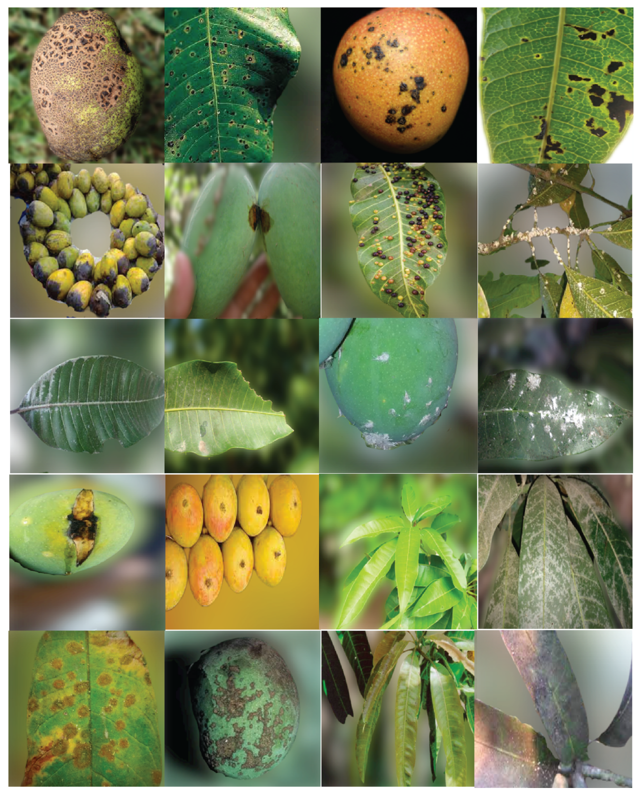

Figure 2. The figure illustrates some of the images of our dataset.

new and unique training examples from the training data. If the dataset used by the machine learning model is extensive and sufficient, the model performs better and is more accurate. Image data augmentation is the most well-known type of data augmentation, and it involves changing images from the training dataset into modified duplicates that belong to the same class as the original image. Transforms include shifts, flips, zooms, and other operations that modify images. Geometric transformation, color transformation, rotation reflection transformation, noise injection, and other conventional data augmentation techniques are only a few examples. A variety of strategies are widely applied in model training. The performance of the model can also be improved by conventional data augmentation techniques and other technologies, according to a number of articles, but the most fundamental geometric transformation stands out as being the most significant of all [21][22]. We extended our dataset using the TensorFlow-based data augmentation approach employing brightness enhancement, rotation, flip, and hybrid augmentation.

C. Deep Learning Models

In this subsection, we have described the different deep-learning methods that we have used in our work.

1) DenseNet169

An example of a convolutional neural network known as a DenseNet is one that employs Dense Blocks, which directly connect all levels, to provide dense connections between layers. A convolutional neural network has three essential layers: a convolutional layer, a pooling layer, and a fully connected layer. Each layer takes additional inputs from all lower levels in order to retain the feed-forward nature, and it transmits its feature mappings to all higher layers.
The network is organized into numerous densely linked dense blocks in the densenet169 design to assist down-sampling; the architecture of densenet169 is shown in Figure 3. Transition layers are the layers between blocks that perform convolution and pooling. In our experiments, a batch normalization layer is followed by an 11 convolution layer and a 22 average pooling layer[23][24].

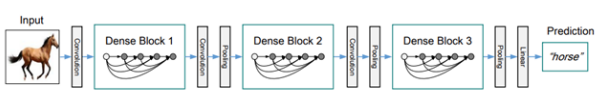

Figure 3. Architecture of Densenet169 [25].

2) Inception V3

A deep learning model for categorizing images that is based on convolutional neural networks is called Inception V3. The core model Inception V1, which was released in 2014 as GoogLeNet, was upgraded with the Inception V3. [26]. The 42 layers and the lower error rate of the Inception v3 model, which was introduced in 2015, distinguish it from its forerunners. [27].

3) MobilenetV2

A convolutional neural network architecture designed specifically for mobile devices is called MobileNetV2. It is based on a residual structure that is inverted and has residual links between bottleneck levels. The intermediate expansion layer uses light depth-wise convolutions to filter features as a source of non-linearity. In the MobileNetV2 design, there is a 32-filter initial fully convolution layer followed by 19 further bottleneck layers.[26].

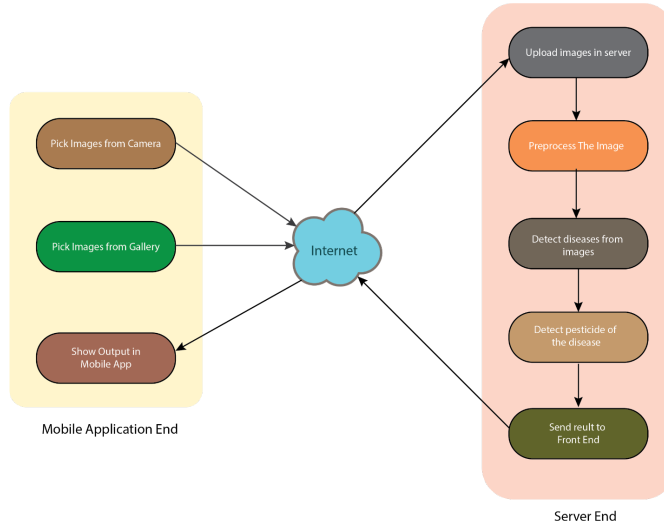

Figure 4. Front-end and back-end flow of mobile app.

D. Mobile App development

We developed an application in our mobile app development section that can determine disease using images from the camera and gallery and suggest pesticides according to the disease. This project's work is divided into two phases: front-end development and back-end development. For the front end of our application, we used React-native, and for the back end, we used NodeJs.

1) Front-end development

Real-time, natively generated mobile apps for iOS and Android may be made using the JavaScript framework React Native. The user has two options in this application: one is to capture images from the camera, and the other is to choose images from the gallery. Regarding the selection of images, the image is delivered to the backend. The detection result and pesticide recommendation for the detected disease are provided from the backend. The application then displays the results to the user.

2) Back-end development

The server gets the detection request with an image in the backend. The server next does preprocess followed by detection using the best methods we've learned from the results and an evaluation section on the image. Once the detection result is obtained, the server identifies the pesticides according to the disease. It delivered the output results to the users' end through the internet when it had completed all of the prerequisites. NodeJs was used to do all of this work. NodeJs is a JavaScript-based framework for building single-page apps, video streaming websites, online chat applications, and other I/O-demanding web applications. The Figure 4 shows the flow of our application.

4. EVALUATION METRICES

To evaluate performance of the algorithms, the confusion matrix is useful for quickly measuring precision and recall

TABLE III. Results of different Models

MODEL	Training Accuracy	Validation Accuracy	Precision	Recall	F1-Score
INCEPTION V3	0.976	0.9798	0.95	0.94	0.93
DENSENET169	0.9781	0.9937	0.97	0.96	0.96
MOBILENETV2	0.975	0.9812	0.95	0.93	0.93

given predicted and real labels from a model.

Unlike True Negative (TN), which denotes a negative item that is anticipated to be in the negative class, True Positive (TP) denotes a positive item that is anticipated to be in the positive class. False Positive (FP) is the term for a negative class item that is expected to be in the positive class. In contrast, False Negative (FN) is the term for a positive class entity that is anticipated to be in the negative class.

The ratio of correctly predicted objects to the total number of objects is used to measure accuracy. From the confusion matrix, we may determine it as follows:

$$Accuracy = \frac{number of correctly predicated object}{total number of object} \quad (1)$$

$$Accuracy = \frac{TN + TP}{TP + FP + FN + TP} \quad (2)$$

Precision refers to how many of the picked things are relevant. Precision may be calculated using the confusion matrix as follows:

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

Recall [28] [29] refers to how many relevant items are chosen. As a result of the confusion matrix, it may be expressed as follows:

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

5. Result & Discussion

On our build dataset, we have applied different algorithm such as InceptionV3, MobileNetV2, and DenseNet169. Using these algorithms get produce training accuracy, Validation accuracy, Precision, Recall, and F1-score which is given below table. For inceptionV3 training accuracy 97.60%, validating accuracy 97.98%, Precision 95%, Recall 94%, and F1-Score 93%. For DenseNet169 training accuracy 97.81%, validating accuracy 99.37%, Precision 97%, Recall 96%, and F1-Score 96%. For inceptionV3 training accuracy 97.50%, validating accuracy 97.12%, Precision 95%, Recall 93%, and F1-Score 93%.

In Table III, the results of different models is shown. In Figure 5, shows the comparison between different models .

Figure 6, displays the InceptionV3 confusion matrix. Our dataset was essentially divided 70-30 for training and testing. According to the InceptionV3 confusion matrix, this model achieves the highest true positive value (1) in six of the classes, the average value is greater than (0.9) in thirteen classes, and the worst value (0.42) is only attained in one class. Figure 7 displays the Densenet169 confusion matrix, which demonstrates that this model achieves the highest true positive value (1) in ten (10) of the classes, In nine classes, the average value is higher than (0.9), with the typical leaf class being detected yielding the poorest score (0.58). The confusion matrix for MobilenetV2 is shown in Figure 8, where it can be seen that the Normal Leaf class is detected with the lowest true positive value (0.44). Only four (4) of the classes yield the highest real positive value (1).

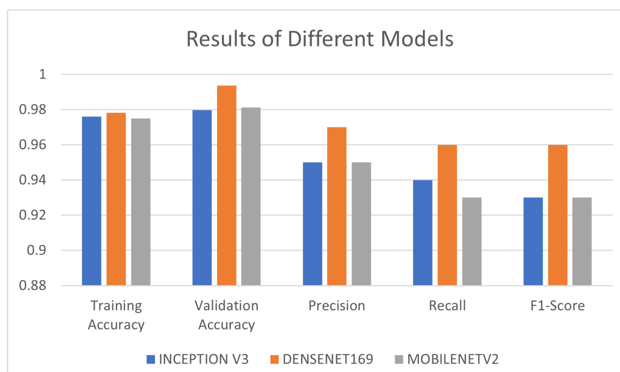

Figure 5. Result of Different Classes.

From Table IV, we have seen that most of the reviewed papers use their own dataset. They deploy different methods like SVM, AlexNet, CNN, DenseNet201, K-means etc in their research. Most of the research has been done over 4,5 or 6,7 or 9 classes. But here we have worked with 20 classes which are much larger compared to the related works. We have also developed a mobile application where the user easily uploads their image and identifies the disease and a pesticide suggestion is also available for that disease. We used our own dataset for this research. We were unable to locate a dataset that was comparable in terms of mango leaf and fruit diseases with pesticide recommendations. The majority of the work was classified in binary terms, such

TABLE IV. Comparison between Our work and others work.

Paper no	Own Dataset (Yes/No)	Techniques	Accuracy	Mobile App	Classes	Leaf	Fruits
[5]	Yes	SVM	90%	No	4	Yes	Yes
[2]	No	AlexNet	98.30%	No	9	Yes	No
[3]	Yes	CNN	98.60%	Yes	1	No	Yes
[6]	Yes	CNN	70%	No	4	No	Yes
[7]	Yes	K-means	90%	No	4	Yes	No
[8]	Yes	SVM	68.89%	No	3	Yes	No
[9]	Yes	NNE	80%	No	7	Yes	No
[30]	Yes	SVM	94.13%	No	6	No	Yes
[10]	Yes	ANN	89.41%	No	4	Yes	No
[11]	Yes	RBFNN	91.15%	No	4	Yes	No
[13]	Yes	DENSENET201	98%	No	5	Yes	No
[5]	Yes	CNN	96.16%	No	4	Yes	No
[31]	Yes	MCNN	97.13%	No	4	Yes	No
[14]	Yes	CNN	96.67%	No	5	Yes	No
Proposed method	**Yes**	**DenseNet-169**	**97.81%**	**Yes**	**20**	**Yes**	**Yes**

TABLE V. Accuracy Comparison among Different Methods

Method Name	Accuracy
CNN	91.32%
AlexNet	93.06%
Densenet-201	92.09%
Inception V3	97.60%
MobileNet V2	97.52 %
Proposed Method: DenseNet- 169	97.81%

as healthy or unhealthy, using only fruit or only mango. The majority of the works also didn't take the pesticide recommendations into account. Therefore, we applied some of the methods in our dataset that were used in the related works, and the accuracy we obtained is presented in TableV. TableV makes it quite evident that the method we selected, DenseNet169, has the highest accuracy.

6. Mobile Application Implementation & Evaluation

After completing the construction of our application, we tested it with a variety of disease and healthy fruit and leaf photos. In the vast majority of cases, our application predicted accuracy as accurately as we had hoped. The pesticide recommendation can also be noticed in the application. Figure 9 shows several application screenshots.

For the evaluation of our mobile application, we have taken random images from different classes with a fixed number of samples (20) and tested the average detection time and it presents a promising result. TableVI shows the average speed for detecting mango diseases.

TABLE VI. Average Speed Detection time in Mobile Application

Class Name	No of Samples	Average Speed Detection Time
Anthracnose Fruit	20	2.1s
Anthracnose Leaf	20	2.15s
Bacterial Canker Fruit	20	1.58s
Bacterial Canker Leaf	20	2.23s
Black Rot Disease Fruit	20	2.15s
Fruit Borer	20	2.13s
Gall Midge Leaf	20	2.22s
Giant Mealybug Leaf	20	1.56s
Coating Mite Leaf	20	2.3s
Cutting Weevil Leaf	20	2.5s
Mealybug Fruit	20	2.45s
Mealybug Leaf	20	2.25s
Mysterious Cracking Disease Fruit	20	2.15s
Normal Fruit	20	1.46s
Normal Leaf	20	2.17s
Powdery Mildew Leaf	20	2.2s
Red Rust Leaf	20	1.49s
Scab Fruit	20	2.56s
Scab Leaf	20	2.38s
Shoot Mold Leaf	20	2.29s

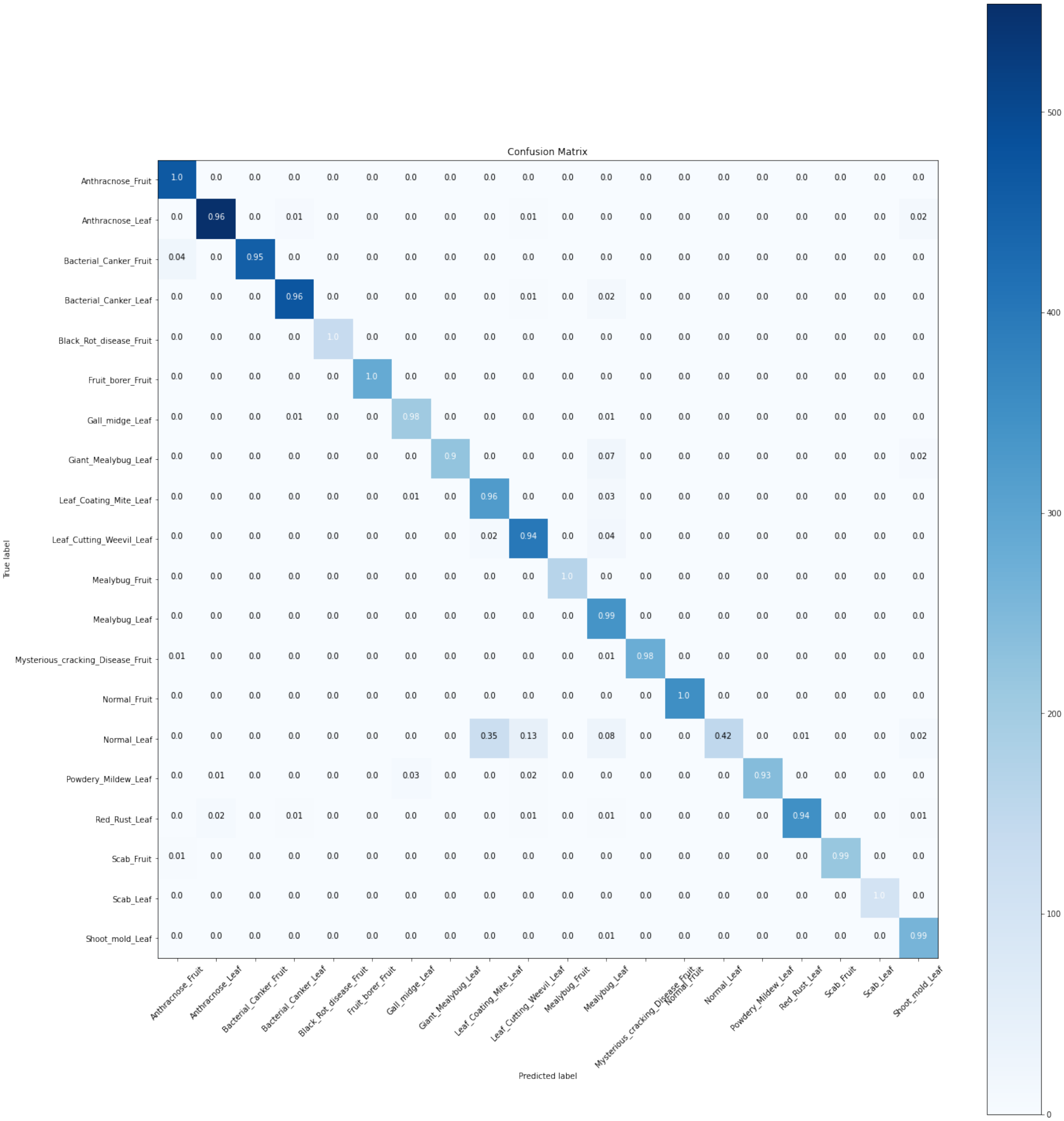

Figure 6. Confusion Matrix of InceptionV3

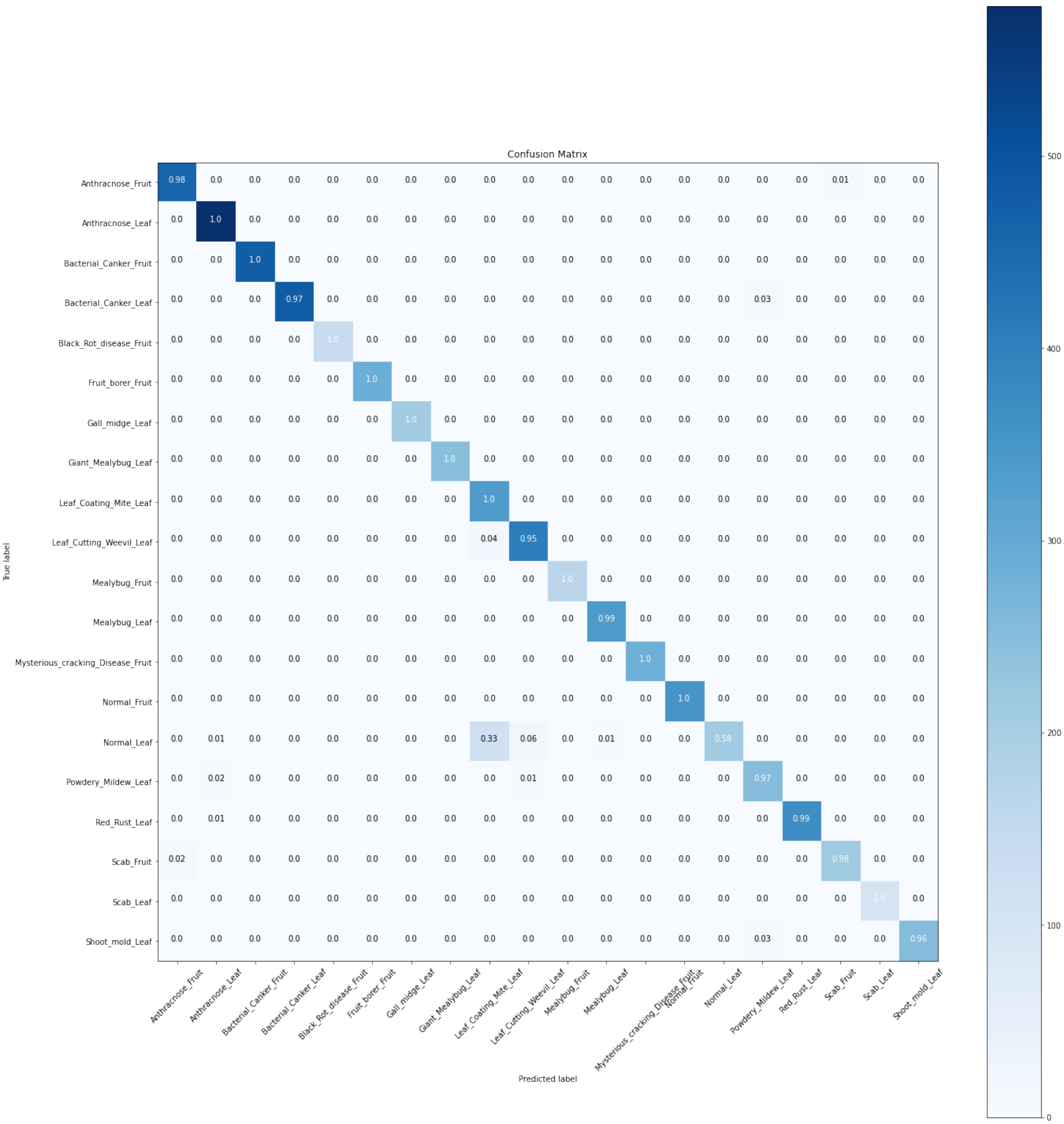

Figure 7. Confusion Matrix of DenseNet196

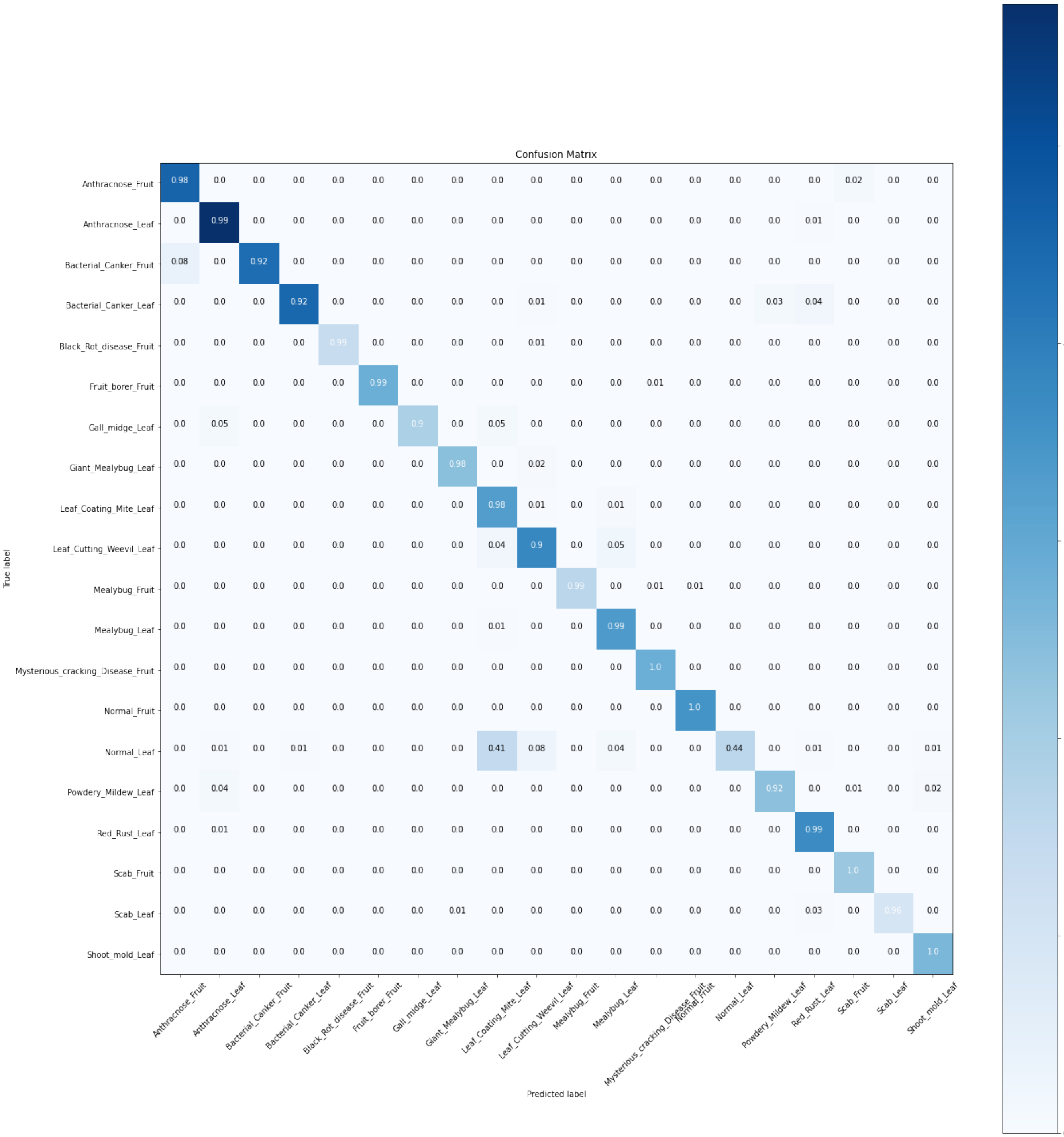

Figure 8. Confusion Matrix of MobileNetV2

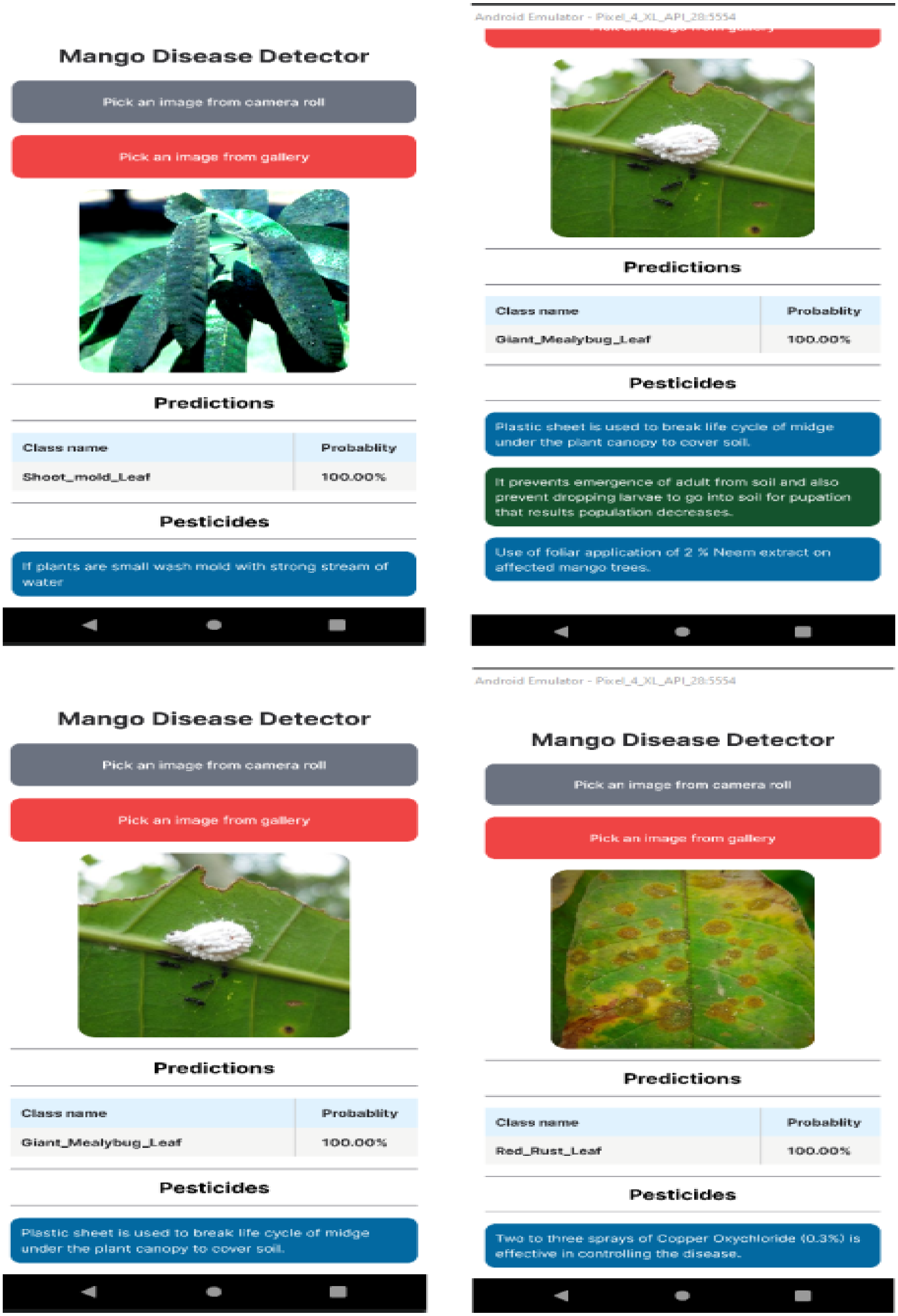

Figure 9. Application Screenshot

7. Conclusions and Future Work

Most importantly ensuring diseases and insects free tree plants is a very challenging task. Early detection of diseases and insects with their solution can be the key to avoiding the demolition of mango plants. This paper introduced new datasets of diseases, insects, and their solutions. It includes a sufficient number of mango tree plants, insects, and diseases datasets with pesticide suggestions. This paper also proposed a new machine learning to classify diseases, insects, and their solutions.We used the InceptionV3, MobileNetV2, and DenseNet169 models in our research. DenseNet169 seems to have the best accuracy of 97.81% among them. We developed a mobile application that saves a lot of time, money, and makes use of technology that would be advantageous to the consumers. By uploading images to the application, we evaluated different machine-learning methods to automatically detect the symptom of the leaf and fruit diseases. Though we have built a dataset that includes quality images but the number of images can be increased more. Although we have used certain image-enhancing techniques to boost the quality of the low-quality images, we can still see room for improvement in our model's detection rate when dealing with low-quality images. In the future, we want to expand our dataset. We will also add live camera-based disease detection in our application.

ACKNOWLEDGEMENTS

We would like to thank Bangladesh University of Business & Technology and AML lab for their suggestions and resource sharing.

References

[1] L. Reddy, Y.-J. Wee, W. Ye, and M. Korivi, "Nutritional composition and bioactive compounds in three different parts of mango fruit," *International Journal of Environmental Research and Public Health*, vol. 18, p. 741, 01 2021.

[2] S. Arya and R. Singh, "A comparative study of cnn and alexnet for detection of disease in potato and mango leaf," 09 2019, pp. 1–6.

[3] V. Ashok and V. D S, *A Novel Fusion of Deep Learning and Android Application for Real-Time Mango Fruits Disease Detection*, 01 2021, pp. 781–791.

[4] G. Tumang, "Pests and diseases identification in mango using matlab," 07 2019, pp. 1–4.

[5] S. Veling, "Mango disease detection by using image processing," *International Journal for Research in Applied Science and Engineering Technology*, vol. 7, pp. 3717–3726, 04 2019.

[6] S. Wongsila, P. Chantrasri, and P. Sureephong, "Machine learning algorithm development for detection of mango infected by anthracnose disease," 03 2021, pp. 249–252.

[7] J. Jos and V. Kanyakumari, *Detection of Disease in Mango Trees Using Color Features of Leaves*, 03 2020, pp. 583–593.

[8] R. A. Jm Gining, S. Fauzi, N. Yusoff, T. R. Razak, M. H. Ismail, R. Maskat, R. Ahmad, N. Mohd Zaki, and F. Abdullah, "Harumanis mango leaf disease recognition system using image processing technique," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 23, 06 2021.

[9] M. Mia, S. Roy, S. Das, and M. Rahman, "Mango leaf disease recognition using neural network and support vector machine," *Iran Journal of Computer Science*, vol. 3, 09 2020.

[10] T. Pham, L. Tran, and S. Dao, "Early disease classification of mango leaves using feed-forward neural network and hybrid metaheuristic feature selection," *IEEE Access*, vol. 8, pp. 189 960 – 189 973, 10 2020.

[11] S. Chouhan, D. U. Singh, and S. Jain, "Web facilitated anthracnose disease segmentation from the leaf of mango tree using radial basis function (rbf) neural network," *Wireless Personal Communications*, vol. 113, 07 2020.

[12] A. Rajbongshi, T. Khan, M. Rahman, A. Pramanik, S. Md, S. Siddiquee, and N. Chakraborty, "Recognition of mango leaf disease using convolutional neural network models: A transfer learning approach," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 23, pp. 1681–1688, 09 2021.

[13] P. Kumar, S. Ashtekar, S. Jayakrishna, B. K P, P. Vanathi, and R. Muthu, "Classification of mango leaves infected by fungal disease anthracnose using deep learning," 04 2021, pp. 1723–1729.

[14] S. Arivazhagan and S. V. Ligi, "Mango leaf diseases identification using convolutional neural network," *International Journal of Pure and Applied Mathematics*, vol. 120, no. 6, pp. 11 067–11 079, 2018.

[15] P. Jawade, D. Chaugule, D. Patil, and H. Shinde, *Disease Prediction of Mango Crop Using Machine Learning and IoT*, 01 2020, pp. 254–260.

[16] A. Rocha and J. Lagarteja, "Philippine carabao mango pest identification using convolutional neural network," 03 2021.

[17] N. Sutrodhor, M. R. Hussein, M. Firoz, P. Karmokar, and T. Nur, "Mango leaf ailment detection using neural network ensemble and support vector machine," *International Journal of Computer Applications*, vol. 181, pp. 31–36, 08 2018.

[18] M. Türkoğlu and D. Hanbay, "Plant disease and pest detection using deep learning-based features," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 27, no. 3, pp. 1636–1651, 2019.

[19] S. C. Nelson, "Mango powdery mildew," 2008.

[20] S. Nelson, "Mango powdery mildew caused by oidium mangiferae," Sep 2013. [Online]. Available: https://www.flickr.com/photos/scotnelson/9808512885

[21] J. Shijie, W. Ping, J. Peiyi, and H. Siping, "Research on data augmentation for image classification based on convolution neural networks," in *2017 Chinese automation congress (CAC)*. IEEE, 2017, pp. 4165–4170.

[22] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," *arXiv preprint arXiv:1712.04621*, 2017.

[23] Kaleido, "Remove background from image." [Online]. Available: https://www.remove.bg/

[24] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4700–4708.

[25] "Kaggle: Your machine learning and data science community," https://www.kaggle.com/, (Accessed on 11/15/2022).

[26] Q. Zou, Y. Cao, Q. Li, C. Huang, and S. Wang, "Chronological classification of ancient paintings using appearance and shape features," *Pattern Recognition Letters*, vol. 49, pp. 146–154, 2014.

[27] PyTorch, "Densenet-169," Dec 2017. [Online]. Available: https://www.kaggle.com/pytorch/densenet169

[28] "Alexnet - imagenet classification with convolutional neural networks," Nov 2018. [Online]. Available: https://neurohive.io/en/popular-networks/alexnet-imagenet-classification-with-deep-convolutional-neural-networks/

[29] B. Juba and H. S. Le, "Precision-recall versus accuracy and the role of large data sets," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 4039–4048.

[30] M. Mia, A. C. Chhoton, M. Mozumder, S. Hossain, and A. Hossan, "An approach for mango disease recognition using k-means clustering and svm classifier," 11 2019, pp. 404–409.

[31] D. U. Singh, S. Chouhan, S. Jain, and S. Jain, "Multilayer convolution neural network for the classification of mango leaves infected by anthracnose disease," *IEEE Access*, vol. 7, 03 2019.

Md. Naimur Rahaman is a software engineer and researcher. He received his Bachelor of Engineering degree from the Bangladesh university of Business and technology. He has interests in the fields of Machine Learning (ML), Blockchain, Computer Vision (CV), Virtual Reality (VR), and software development.

Musthafezur Rahaman Chowdhury received his Bachelor of Engineering degree from the Bangladesh University of Business and Technology (BUBT). He intends to work on automating the identification of most flowers and fruits in his flora. He has interests in the fields of Machine Learning (ML), Image Processing, and Software Quality Assurance (SQA).

Md. Ashiqur Rahman completed his B.Sc. in Computer Science and Engineering from Chittagong University of Engineering and Technology (CUET) in 2017. He is currently working as an Assistant Professor in the department of Computer Science and Engineering at the Bangladesh University of Business and Technology since November 2017, while he is pursuing his MSc at the Military Institute of Science and Technology (MIST). Artificial Intelligence (AI), Machine Learning, Deep Learning, Wireless Networking, and the Internet of Things are among his research interests. His recent interests also explored other related domains in Computer Vision including medical image processing for contributing to the biomedical community.

Humayra Ahmed is currently working as an Assistant Professor in the Department of Computer Science and Engineering of the Bangladesh University of Business and Technology (BUBT) since February 2018. She received her B.Sc. in Computer Science and Engineering from Chittagong University of Engineering and Technology (CUET) in the year 2017. Her research interests include artificial intelligence (AI), Machine Learning (ML), and Cognitive Radio (CR). She is currently pursuing her MSc at the Military Institute of Science and Technology (MIST).

Md. Mamun Hossain is currently working as an Assistant Professor in the Department of Computer Science and Engineering of the Bangladesh University of Business and Technology (BUBT). He completed B.Sc. (Engg.) in Computer Science and Engineering from the University of Chittagong, in 2017. He is continuing his M.Sc. in Computer Science and Engineering, at the Military Institute of Science and Technology (MIST). His research interests include Artificial Intelligence (AI), Machine learning, Deep learning, and the Internet of Things. His recent interests also explored other related domains in Computer Vision including medical image processing for contributing to the biomedical community. He is also working as a Member Secretary of the Thesis/Project Management and Distribution committee in his department.

Md. Hasibur Rahman received the B.Sc degree in Computer Science and engineering from Rajshahi University of Engineering & Technology (RUET) in 2017. He worked as a lecturer of Computer Science and Engineering at Bangladesh University of Business and Technology from 2018 to 2022. Currently he is pursuing his PhD.

Md. Shovon Biswas is working as a Software Engineer (Python Django).He has finished his Bachelor of Engineering degree from Bangladesh University of Business and Technology (BUBT). He has interest in Machine Learning (ML), Blockchain, Computer Vision (CV).

Md. Abdul Kader is working as a Frontend Engineer. He is passionate about building scalable and maintainable web applications. He has finished his Bachelor of Engineering degree from Bangladesh University of Business and Technology (BUBT). His research interest includes Machine Learning (ML), Blockchain, Computer Vision (CV).

Tanvir Ahmad Noyan received his Bachelor of Engineering degree from the Bangladesh University of Business and Technology. He is inclined toward the field of Data Science and is interested to work in that field of science.He has proficiency in Data Science with a focus on Feature Engineering, Data Preprocessing, Front-end Software Development, JavaScript, and python programming language.

Milon Biswas is an Information Technology graduate from Jahangirnagar University, Bangladesh (Member, IEEE). He worked as Assistant Professor and Coordinator of Diploma holders of the Department of Computer Science and Engineering at the Bangladesh University of Business and Technology (BUBT) from 2018 to 2022. His research focuses Artificial intelligence, machine learning, health informatics, medical image processing, and data mining with applications. He has multiple publications and was awarded the National Science and Technology Fellowship by ICT Ministry of People's Republic of Bangladesh for his research work during graduate studies. He has joined as a Ph.D. student in Computer and Information Science Department at University of Alabama Birmingham from August 2022.