MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR
(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

Project Report

on

Plant Pathology

Submitted by:
Shubham Kumar Vaish
0901CS191119

Shreya Dhanoliya
0901CS191117

Faculty Mentor:
Mahesh Parmar
Assistant Professor, MITS Gwalior

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE
GWALIOR - 474005 (MP) est. 1957

MAY-JUNE 2022

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR
(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

Project Report

on

Plant Pathology

A project report submitted in partial fulfilment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

Submitted by:

Shubham Kumar Vaish

0901CS191119

&

Shreya Dhanoliya

0901CS191117

Faculty Mentor:

Mahesh Parmar

Assistant Professor, MITS Gwalior

Submitted to:

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE
GWALIOR - 474005 (MP) est. 1957

MAY-JUNE 2022

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

CERTIFICATE

This is certified that **Shreya Dhanoliya 0901CS191117 has** submitted the project report titled **Plant Pathology** under the mentorship of **Assistant Professor Mahesh Parmar,** in partial fulfilment of the requirement for the award of degree of Bachelor of Technology in Computer Science and Engineering from Madhav Institute of Technology and Science, Gwalior.

09/05/2022

Mr. Mahesh Parmar
Assistant Professor
Faculty Mentor
MITS Gwalior
Computer Science and Engineering

Dr. Manish Dixit
Professor and Head,
Computer Science and Engineering

Dr. Manish Dixit
Professor & HOD
Department of CSE
M.I.T.S. Gwalior

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

CERTIFICATE

This is certified that **Shubham Kumar Vaish 0901CS191119** has submitted the project report titled **Plant Pathology** under the mentorship **of Mahesh Parmar, Assistant Professor, MITS Gwalior Computer Science and Engineering** in partial fulfilment of the requirement for the award of degree of Bachelor of Technology in Computer Science and Engineering from Madhav Institute of Technology and Science, Gwalior.

09/05/2022

Mr. Mahesh Parmar
Assistant Professor
Faculty Mentor
MITS Gwalior
Computer Science and Engineering

Dr. Manish Dixit
Professor and Head,
Computer Science and Engineering

Dr. Manish Dixit
Professor & HOD
Department of CSE
M.I.T.S. Gwalior

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR
(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

DECLARATION

We hereby declare that the work being presented in this project report, for the partial fulfilment of requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering at Madhav Institute of Technology & Science, Gwalior is an authenticated and original record of my work under the mentorship **of Mahesh Parmar, Assistant Professor, MITS Gwalior, Computer Science and Engineering**.

We declare that we have not submitted the matter embodied in this report for the award of any degree or diploma anywhere else.

Sh. Vaish
Shubham

Shubham Kumar Vaish

0901CS191119

III Year,
Computer Science and Engineering

Shreya

Shreya Dhanoliya

0901CS191117

III Year,
Computer Science and Engineering

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)

ACKNOWLEDGEMENT

The full semester project has proved to be pivotal to my career. I am thankful to my institute, **Madhav Institute of Technology and Science** to allow me to continue my disciplinary/interdisciplinary project as a curriculum requirement, under the provisions of the Flexible Curriculum Scheme (based on the AICTE Model Curriculum 2018), approved by the Academic Council of the institute. I extend my gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

I would sincerely like to thank my department, **Department of Computer Science and Engineering, for allowing** me to explore this project. I humbly thank **Dr. Manish Dixit**, Professor and Head, Department of Computer Science and Engineering, for his continued support during the course of this engagement, which eased the process and formalities involved.

I am sincerely thankful to my faculty mentors. I am grateful to the guidance of **Mahesh Parmar Assistant Professor MITS Gwalior**, Computer Science and Engineering, for his continued support and guidance throughout the project. I am also very thankful to the faculty and staff of the department.

Shubham Kumar Vaish

0901CS191119

3rd Year,

Computer Science and Engineering

Shreya Dhanoliya

0901CS191117

3rd Year,

Computer Science and Engineering

ABSTRACT

In a developing country like India agriculture plays a noteworthy role. Agricultural intervention in the livelihood of rural India indulges by about 58%. Among the agricultural products, Apple is one of the most used fruit. Thus, preventing significant loss in quantity and yield of Apple is majorly dependent on recognition and classification of diseases a Apple plant might possess. Latest and fostering technologies like Image processing is used to rectify such issues using different types of techniques and algorithms. Initially, the leaves of a Apple plant get affected, when plant develops a particular type of disease. In this project, four consecutive stages are used to discover the type of disease. The four stages include pre-processing, leaf segmentation, feature extraction and classification. To remove the noise, we are doing the pre-processing and to part the affected or damages area of the leaf, image segmentation is used. The k-nearest neighbours (KNN) algorithm, which is a guided, supervised and advance machine learning algorithm, is implemented to find solutions for both the problems related to classification and regression. During the terminal stage, user is recommended with the treatment. Mostly live plants are adversely affected by the diseases. This paper imparts representation of leaf disease detection employing image processing that can identify drawbacks in apple plant.

सार:

भारत जैसे विकासशील देश में कृषि एक उल्लेखनीय भूमिका निभाती है। ग्रामीण भारत की आजीविका में कृषि हस्तक्षेप लगभग 58 प्रतिशत है। कृषि उत्पादों में सेब सबसे अधिक उपयोग किए जाने वाले फलों में से एक है। इस प्रकार, सेब की मात्रा और उपज में महत्वपूर्ण नुकसान को रोकना मुख्य रूप से सेब के पौधे के रोगों की पहचान और वर्गीकरण पर निर्भर करता है। विभिन्न प्रकार की तकनीकों और एल्गोरिदम का उपयोग करके ऐसे मुद्दों को ठीक करने के लिए इमेज प्रोसेसिंग जैसी नवीनतम और बढ़ावा देने वाली तकनीकों का उपयोग किया जाता है। प्रारंभ में, एक सेब के पौधे की पत्तियां प्रभावित होती हैं, जब पौधे एक विशेष प्रकार की बीमारी विकसित करता है। इस परियोजना में रोग के प्रकार का पता लगाने के लिए लगातार चार चरणों का उपयोग किया जाता है। चार चरणों में प्री-प्रोसेसिंग, लीफ सेगमेंटेशन, फीचर निष्कर्षण और वर्गीकरण शामिल हैं। शोर को दूर करने के लिए हम पूर्व-प्रसंस्करण कर रहे हैं और पत्ती के प्रभावित या क्षतिग्रस्त क्षेत्र को विभाजित करने के लिए, छवि विभाजन का उपयोग किया जाता है। k-निकटतम पड़ोसी (KNN) एल्गोरिथ्म, जो एक निर्देशित, पर्यवेक्षित और उन्नत मशीन लर्निंग एल्गोरिथम है, को वर्गीकरण और प्रतिगमन से संबंधित दोनों समस्याओं के समाधान खोजने के लिए लागू किया गया है। टर्मिनल चरण के दौरान, उपचार के साथ उपयोगकर्ता की सिफारिश की जाती है। ज्यादातर जीवित पौधे बीमारियों से प्रतिकूल रूप से प्रभावित होते हैं। यह पेपर छवि प्रसंस्करण को नियोजितकरकेपत्तीरोगकापतालगानेकाप्रतिनिधित्वप्रदान करता है जो किसान को तेज और विश्वसनीय परिणाम देने के लिए रंग, बाध्य और बनावट के आधार पर टमाटर के पौधे में कमियों की पहचान कर सकता है।

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1: INTRODUCTION

The production of fruits and crops across the globe is highly influenced by various diseases. A decrease in production leads to an economic degradation of the agricultural industry worldwide. Apple trees are cultivated worldwide, and apple is one of the most widely eaten fruits in the world. The world produced an estimated 86 million tons of apples in 2018, and production and consumption have increased ever since. However, the average national yield of apples is low in comparison to the potential yield of apples. The major factors for the low production of apples are ecological factors, poor post-harvest technologies, less thrust on basic research, inadequate supply of quality planting materials to farmers and socio-economic constraints, etc. Despite their high consumption and medicinal benefits, apple trees are prone to a variety of diseases caused due to insects and micro-organisms such as bacteria. There are several diseases which attack apple, the major one being anthracnose (Neofabraea spp.) cedar apple rust (Gymnosporangium juniperivirginianae), fireblight (Erwinia amylovora), scab (Venturia inaequalis) and powdery mildew (Podosphaera leucotricha). The proper care of trees using fertilizers is thus an important step.

A timely determination of such conditions in the leaves can help the farmers and prevent further losses by taking proper actions. Using just the traditional approaches for diagnosing the plant's disease, farmers often miss the ideal time for preventing such diseases, since the use of these conventional diagnostic approaches takes a lot of time. Currently, there are no automated procedures for such timely detection, and expert supervision is required frequently.

A lack of automation leads to a waste of time and money, which deteriorates the quality of fruits and crops. Advancement in technology has directed machine learning and additional soft computing methods in this domain, which are very useful in the automatic detection and classification of diseases in various plants. Anuradha Badage proposed a system that can periodically inform the farmers in advance about the crop diseases and help them to take the required actions. The system used Canny Edge Detection, which captures the deformities in the leaves and color changes in the leaves to identify the diseases accurately. In another research, Korkut, Umut Baris et al. collected images of leaves of different plant species and extracted their features via transfer learning. After that, various machine learning methods were employed on the extracted features, and the final model was achieved with an accuracy of 92.76%.

Despite the advent of different machine learning approaches to enhance the overall efficiency of disease analysis in plants/crops, multiple factors such as light conditions of crop images and disease variations affect the detection accuracy. An evident advantage of deep learning over machine learning is that deep learning techniques can be applied directly to raw data in various formats such as .csv, .jpg, etc. Machine learning, on the other hand, requires an additional step of pre-processing in the form of feature extraction. Conventional machine learning algorithms such as Support Vector Machines (SVMs), Decision Trees, Bayesian Networks [8], etc., are flat-algorithms. Flat indicates that these algorithms cannot be applied to raw data directly. Figure 1 shows the steps involved in solving a problem via machine learning. The raw data are fed to the model directly in deep learning and any impurity in data can lead the incorrect learning of the model, and thus will not classify it correctly

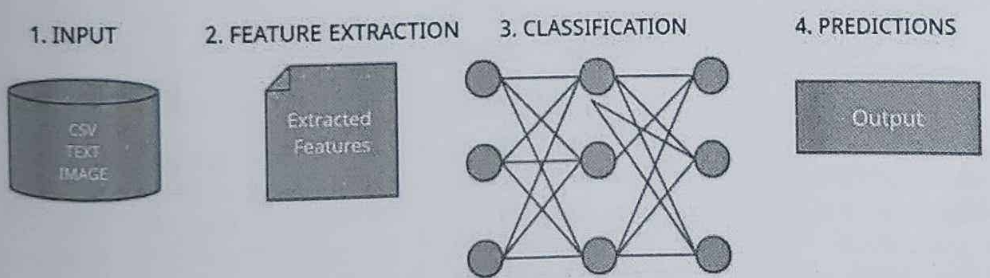


Figure 1. Basic steps involved in machine learning problems.

The primary advantage of using deep learning techniques is to eliminate the need for feature extraction. Feature extraction is a complex process and requires a deep understanding of the problem in hand. Deep learning algorithms determine high-level features from data in an incremental manner. Figures 1 and 2 reveal the steps associated with a deep learning problem. Deep learning combines the steps of feature extraction and classification. Another advantage of deep learning algorithm is their great potential to work with extensive data [10]. Pardede, Hilman F. et al. [11] used a convolutional autoencoder to counter the problem of hand-crafted features. The autoencoder extracted features via unsupervised learning techniques. The outputs of the autoencoder are later fed to an SVM-based classifier for the purpose of feature learning. The advancements in the domain of computer vision and deep learning have led to the use of convolutional neural networks (CNNs).

CNNs have become the go-to models for image classification tasks. The most significant advantage is its architecture and how it extracts and passes features to the subsequent layers. A CNN consists of broadly two modules—the feature extraction module and the classifier module. The role of the feature extraction module is to draw out the relevant features from the image via convolution and pooling. The classifier module acts upon the extracted features and performs the task of output predictions. Figure 3 shows a simplified structure of convolutional neural networks.

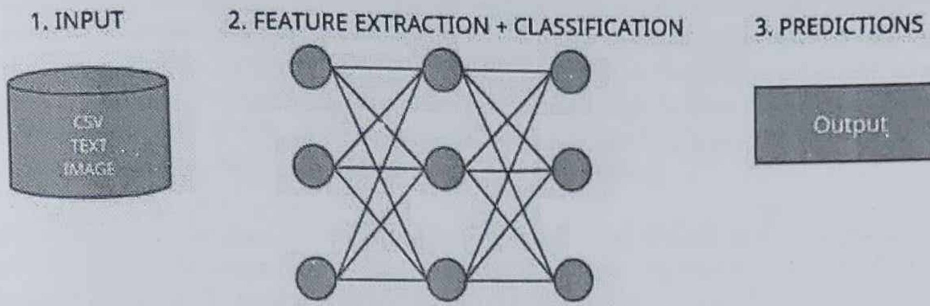


Figure 2. Basic steps involved in deep learning problems.

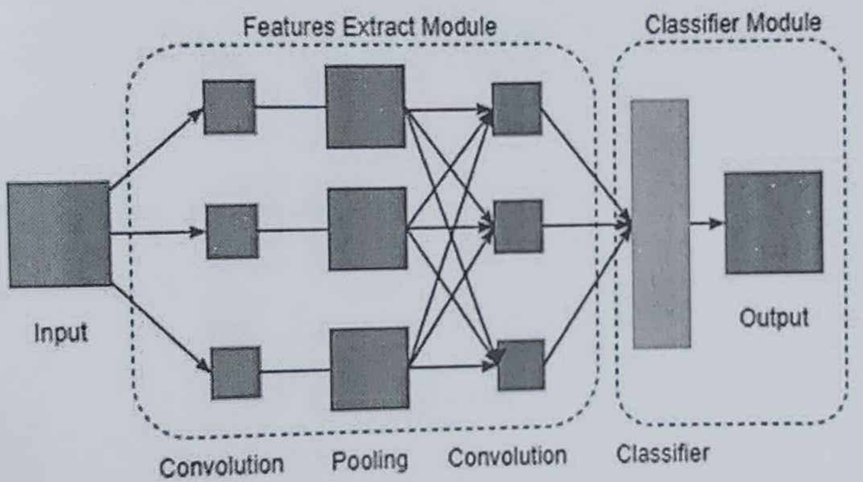


Figure 3. The modules of a basic CNN.

Since 2016, many researchers have started to leverage the potential of CNNs to build better image classifiers. Justine Boulent et al., in their work, summarized 19 different studies which made use of CNNs to detect the diseases in crops automatically. Their work also highlights the significant shortcomings and issues in these studies. Keeping in prospect the success of CNNs as a classifier for image data, we have tried to leverage its functionality for our research. In this paper, we proposed an ensemble of three stateof-the-art deep learning models—DenseNet121, EfficientNetB7, and EfficientNet NoisyStudent to automate the task of disease detection in apple's leaves among four classes—healthy, scab, rust, and multiple diseases.

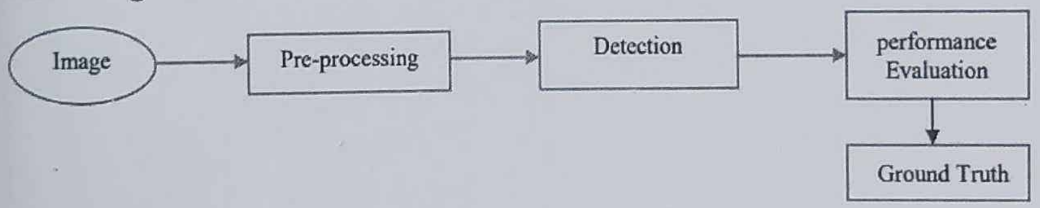
We used transfer learning to transfer the knowledge of the previously learned models into our research. We ensembled the three model's prediction outputs by Model Averaging [18], which reduces the variance observed in the predictions across models. The accuracy achieved by our proposed model on the validation dataset is 92.76%. From the results, it is seen that the model outperformed various other previous models proposed earlier in terms of its performance metrics such as accuracy, etc. The proposed model uses Image Augmentation techniques such as Canny Edge Detection [4], Flipping, Blurring, etc., to increase our dataset's size and develop a more robust and generic model. To the best of our knowledge, the techniques proposed in this paper are not available in the previous literature and can significantly boost the model's performance by providing an enhanced dataset for training. Ensembling has led to a reduced variance in the predictions and produced better accuracy in difficult cases of multiple diseased leaves. In addition to this, the proposed model is deployed using a web application to make it easily accessible for farmers. The remainder of this paper is organized as follows: In Section 2, a detailed background study is presented, which includes the work carried out previously in this domain. Section 3 discusses the proposed methodology, which summarizes the working steps taken to reach our proposed model. Details about the dataset, Image Augmentation techniques used, Modelling, and Ensembling are discussed in this section. Section 4 describes the results obtained and a brief comparison with previous researches. Finally, this paper is concluded in Section 5.

1.1 Objectives and Scope

Plant diseases cause a major production and economic losses in the agricultural industry. The disease management is a challenging task. Usually the diseases or its symptoms such as coloured spots or streaks are seen on the leaves of a plant. In plants most of the leaf diseases are caused by fungi, bacteria, and viruses. The diseases caused due to these organisms are characterized by different visual symptoms that could be observed in the leaves or stem of a plant. Usually, these symptoms are detected manually. With the help of image processing, Automatic detection of various diseases can be detected with the help of image processing. Image processing plays a crucial role in the detection of plant diseases since it provides best results and reduces the human efforts.

The image processing could be used in the field of agriculture for several applications. It includes detection of diseased leaf, stem or fruit, to measure the affected area by disease, to determine the colour of the affected area. Tomato cultivation is one of the most remunerative farming enterprises in India. The naked eye observation by the experts is approach usually taken in identification and detection of plants. This approach is time consuming in huge farms or land areas. The use of image processing techniques in detection and identification of tomato plant diseases in the earlier stages and thereby the quality of the product could be increased. These systems monitor the plant such as leaves and stem and any variation observed from its characteristic features, variation will be automatically identified and also will be informed to the user.

block diagram:



3. Flow Diagram

1.2 Project Features

A symptom of plant disease is a visible effect of disease on the plant. Symptoms may include a detectable change in colour, shape or function of the plant as it responds to the pathogen. Leaf wilting is a typical symptom of verticillium wilt, caused by the fungal plant pathogens Verticillium albo-atrum and V.

Deep learning solves different short comes of machine learning feature extraction such as extracting features manually by using the best and robust technique called a CNN. The layers are used to learn the knowledge. With the use of filtering mechanism, the data are used to match and extract their values.

1.3 Feasibility

The used dataset partitioning technique is K-fold cross-validation which is partitioned as K values, where K + 1 have to be obtained for the upcoming divisions. For this research, the study researcher has assigned the K value as 10 because it is recommended for deep learning [8, 20]. Therefore, K = 10 means 10-fold cross-validation, so dividing the total dataset into 10. D = 2400/10 = 240 data for each fold are used. From this routine activity, 80% (2160 leaf images) yield the most appropriate performance which are trained and rest 20% (240 leaf image) are used for testing; thus, the system was validated.

To collect cotton leaf images for this research, two image capturing devices were used such as a smartphone and digital camera. The proposed model was implemented using python version 3.7.3 for its usages. Also, the model is trained on the deep learning package called Keras, Version: 2.2.4-tf TensorFlow backed. TensorFlow, Version: 1.14.0 was recommended to adopt the proposed system. To evaluate the performance, many experimental setups were conducted with the help of a graphical user interface using Tkinter. From hardware, training and test was carried out on CPU instead of GPU.

1.4 System Requirement

Laptop, PC
Intel i3 or above >2.0 GHz
8GB RAM
Graphics Card 4GB

Mobile Phone
3GB RAM
32 ROM

Chapter 2: Literature Review

Outlined the different formats of graph convolutional neural network. It was prepared to process the uniform electro encephalography data for predicting the four classes of motor imaginaries to relate with electro encephalography electrode. They addressed their data with the transformation of 2D to 3D perspectives. The structure was processed through these dimensional units.

A study stated that, to utilize the dynamic route of deep learning, they proposed short-term voltage stability. They managed the clustering algorithm to obtain short-term voltage stability to increase the reliability.

In it is stated that deep learning technique was applied to identify the leaf diseases in different mango trees. The researchers used four different leaf diseases from various specimens of mango leaves, where they addressed nearly many datasets. The CNN structure was trained with more than 3600 images, where 80% are used for training and 20% are used for testing. Remaining 600 images were used to find the accuracy and to identify the mango leaf diseases which showed the feasibility of its usage in real-time applications. The classification accuracy can be further increased if more images in the dataset are provided by tuning the parameters of the CNN model.

In , detection of cotton leaves were addressed with image processing. Here, K-means algorithms are used to segment the datasets.

The research showed the identification of diseases in banana plants which infect their leaf. In this research study, 3700 images were used for training, but there is no balanced dataset in each class. Researchers performed different experiments, for example, the training mode by using coloured and grayscale image datasets and by using different dataset splitting techniques. They obtained the best accuracy of 92.76% in coloured image and 80% and 20% training to the validation dataset.

Chapter 3: Preliminary design

3.1 Image acquisition

The dataset used in this research is an openly available dataset, which is a subset of the dataset created by the Plant Pathology and Plant-Microbe Biology Section of the Cornell University [41]. The original dataset contains 3651 high-grade images of apple leaves with various foliar diseases. The images are captured by hand in different scenarios such as variable lighting, different angles, different surfaces, etc., to represent a more general dataset that covers most possibilities.

The dataset used in this research contains 3642 images of apple leaves distributed proportionally among four classes—cedar apple rust, multiple diseases, healthy leaves, and apple scab. The distribution of these four classes is shown in the pie chart represented in Figure 5. Out of the 3642 images available to us, only 5% of the plants have multiple diseases, i.e., having both scab and rust. The other three classes—healthy, cedar apple rust, and apple scab, are equivalent in proportion. The correct classification of apple leaves among the following four categories is the prime objective of this research.

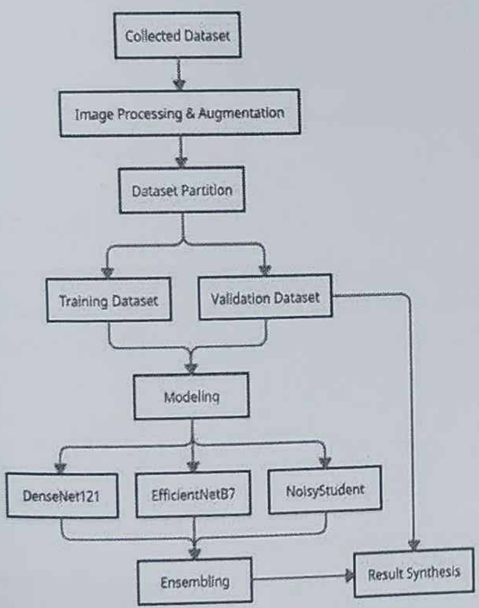
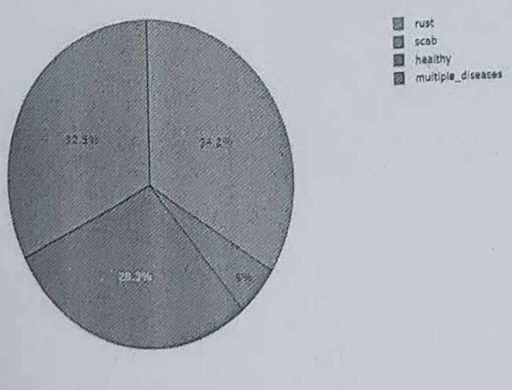


Figure 4. Workflow for the proposed methodology.



5. Distribution of classes

.1.1 Healthy

t can be seen in Figure 6 that healthy leaves are entirely spotless and are green withno signs of any disease. Our lataset contains about 28.3% healthy leaves.

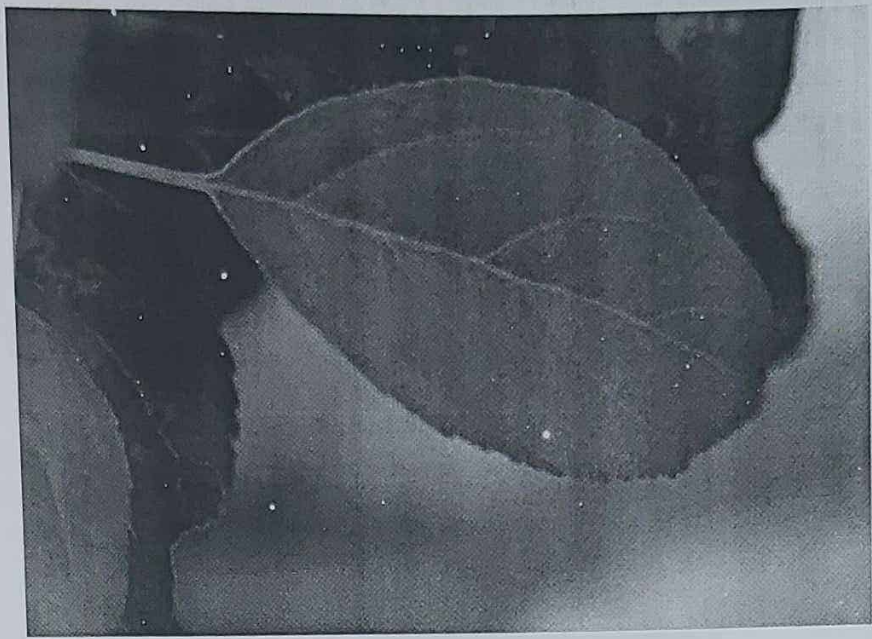


Figure 6. Healthy leaf

3.1.2 Apple Scab

Figure 7 shows the leaf of an apple tree with apple scab disease. We can see that the leaves have brown spots/marks. A scab is often caused by a fungus that infects the leavesand the fruits, which makes the fruit unhealthy for eating. In our dataset, about 32.5% ofimages are of apple scab.

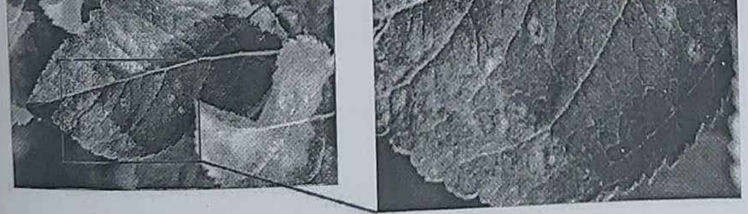


Figure 7. Leaf with apple scab.

3.1.2 Cedar Apple Rust

Figure 8 shows the leaf of an apple tree having cedar apple rust. We can see that the leaves have dense yellowish marks. Rust is often caused in plants via a unique fungus named 'rust fungus'. In our dataset, about 34.2% of images are of cedar apple rust.

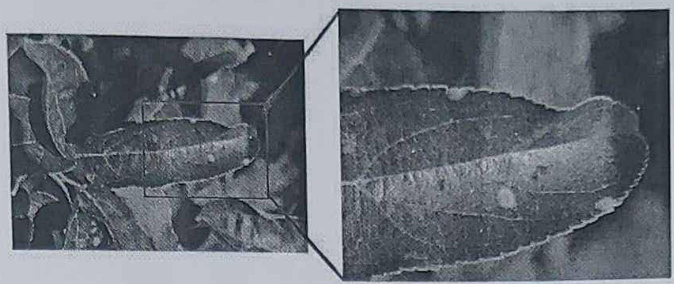


Figure 8. Leaf with cedar apple rust.

3.1.4 Multiple Diseases

Leaves with multiple diseases show signs of having both apple scab, i.e., having brown spots, and cedar apple rust, i.e., having yellow marks as shown in Figure 9. The leaves are severely damaged in this case and are very difficult to treat. Our dataset has only 5% images from this class.

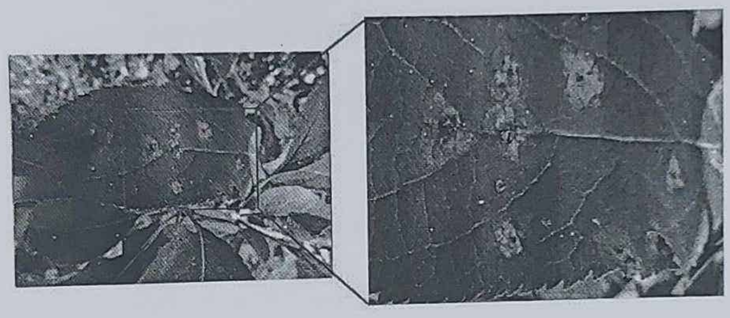


Figure 9. Leaf with multiple diseases.

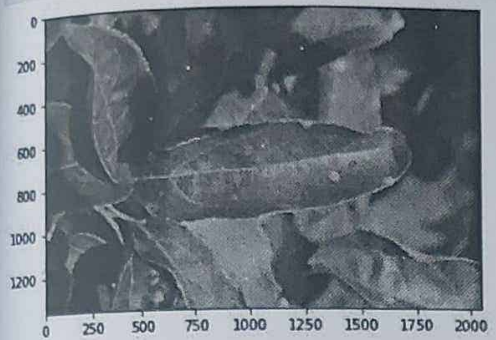
3.2 Colour Transformation

The RGB images were converted into HSV colour space representation. In the RGB model each colour appears in its primary spectral components of red, green and blue and this model is based on a Cartesian co-ordinate system. Though RGB model matches to the human eye in such a way as strongly perceptive to the primary colours, this model is not well suited for describing colours in terms that are practical for human interpretation.

3.2 Segmentation

Separating images into two folders train and test and making 4 sub folders with disease name in train folder

```
image1=Image.open(r'images\Original Dataset\Test_0.jpg')
plt.imshow(image1)
plt.show()
```



10. input

In this work the leaf mask and the damaged area mask thus generated are taken for the further analysis. The Damaged Area Mask was multiplied with the resized RGB image obtained after the pre-processing step. Since the mask contains only 1's and 0's and in the mask the diseased portion of the leaves are having the value of 1's. When this image is multiplied with the RGB image, only the diseased portions will come out as a result of multiplication.

3.4 Detection of the diseased portion

The segmented output image shows the portion of the leaves affected by a disease. Then the total affected area can be calculated by counting the no of pixel values in the output image.

3.5Feature extraction

Feature extraction is the process of defining a set of features, or image characteristics, which will most efficiently or meaningfully represent the information that was important for analysis and classification. In this work statistical techniques are used to describe the textures. In this step Grey Level Co-occurrence matrix of the leaf images are calculated. Grey-level co-occurrence matrix (GLCM) creates a matrix from image a given image I.

3.6 Disease classification

The classification process is done by adopting Neural Network. Back propagation method is considered under the supervised learning mechanism. The feed forward back Propagation Neural Network is generally consisting of three layers such as an input layer, a hidden layer, and an output layer. The neural network is to be trained with the available data.

Chapter 4: Final Analysis and Design

4.1 Results

In this section, the experimental setup is first introduced, followed by the description of various performance metrics, analysis and comparison of our results, a description of computational resources, and information about the model deployment.

4.1.1Experimental Setup

We developed and trained our model using Python3. Along with Python, we lever- aged the functionalities of various frameworks and libraries such as Keras, TensorFlow, Pandas, NumPy for our purpose. We used Kaggle that offers free access to TPUs (Tensor Processing Units) for a fixed number of hours per week. Tensor Processing Units, developed by Google, are custom-developed application-specific integrated circuits (ASICs) to hasten machine learning and deep learning-related work. TPU v3-8 provides 128 GB of RAM and 19.6 GB of disk space along with eight computational cores. We used tf.data dataset api of TensorFlow to create a dataset from the input images and apply transformations to the data. To transform the dataset, we mapped our dataset to functions for Flipping, Rotation, Blurring, etc. We loaded our model using the sequential class of Keras and specified weights as image net. While loading the model, functional calls to the required metrics such as: precision, F1, etc., were selected. We ran around 20 epochs with 12 steps per epochs for a Batch size of 128. The starting value of the learning rate was kept as 1×10^{-5}. The results were obtained via the history object obtained after fitting our models. The visualization of results was achieved by using the Seaborn library offered by Python. Finally, the Ensembling was carried out by averaging the corresponding cell values for the prediction matrices of the trained models.

4.1.2 Performance Metrics

The performance of our proposed model and individual pre-trained models are assessed by four metrics, i.e., Validation Accuracy, positive predictive value (PPV), also known as Precision, Recall, and F1-Score. In addition to these, confusion matrices are shown for each model, and the necessary conclusions are drawn. The plots for Accuracy vs. Epochs have also been presented for the pre-trained models.

Accuracy, in simple words, is the number of accurate predictions made concerning the total predictions made by a model.

$$Accuracy = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

recision is the ratio of the number of correct positive results divided by the number of positive results predicted by the lassifier.

recision = **True Positives + False Positives**

ecall is the ratio of the number of correct positive results divided by the number of all relevant samples (all samples that hould have been identified as positive).

Recall = **True Positives + False Negatives**

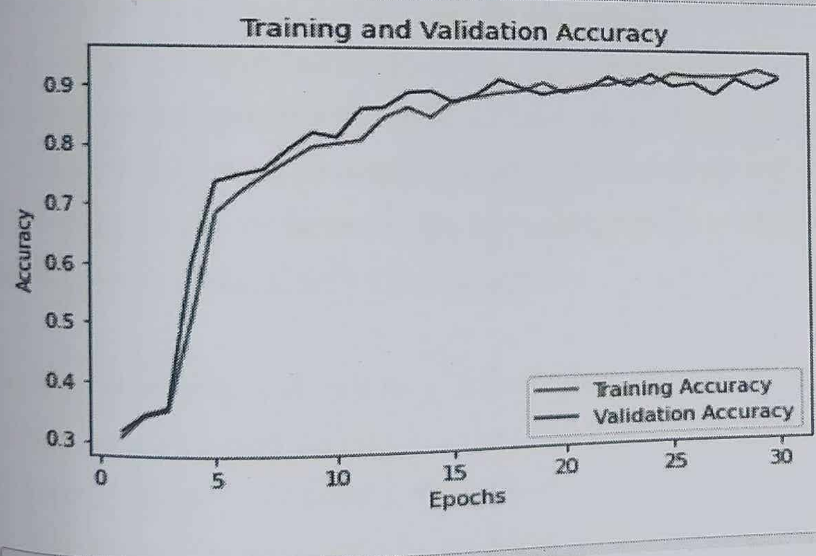
F1-Score is the harmonic mean of Precision and Recall.

$$F1 = \frac{2}{\frac{1}{precision} + \frac{1}{recall}}$$

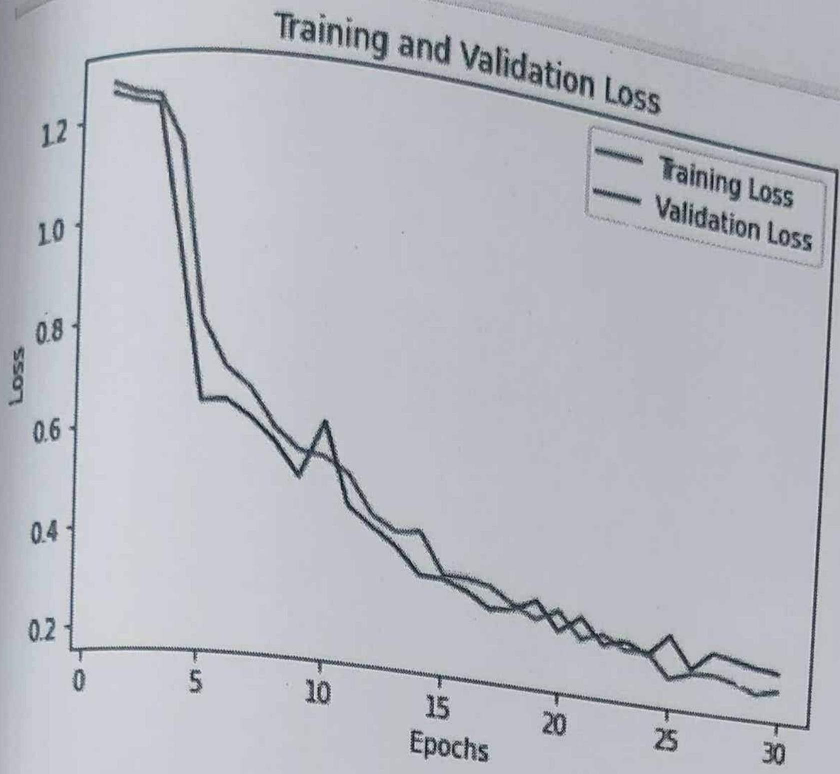
4.1.3Performance Benchmarking

In this section, we present the results of our proposed model alongside the results of the pre-trained models. Table 1 shows the accuracy of the pre-trained models and ourproposed model at different values of the train-test split. It is observed that DenseNet121 and EfficientNetB7 tend to achieve the best accuracy at a 0.15 split ratio, but Efficient NetNoisy Student classifies most accurately at a 0.20 split ratio. Our proposed model achieves an accuracy of 92.76% on a 0.15 split ratio. Figure 11 shows the plots of Accuracy vs. Epochsfor the pre-trained models and our proposed model.

```
plt.show()
```



11 Accuracy

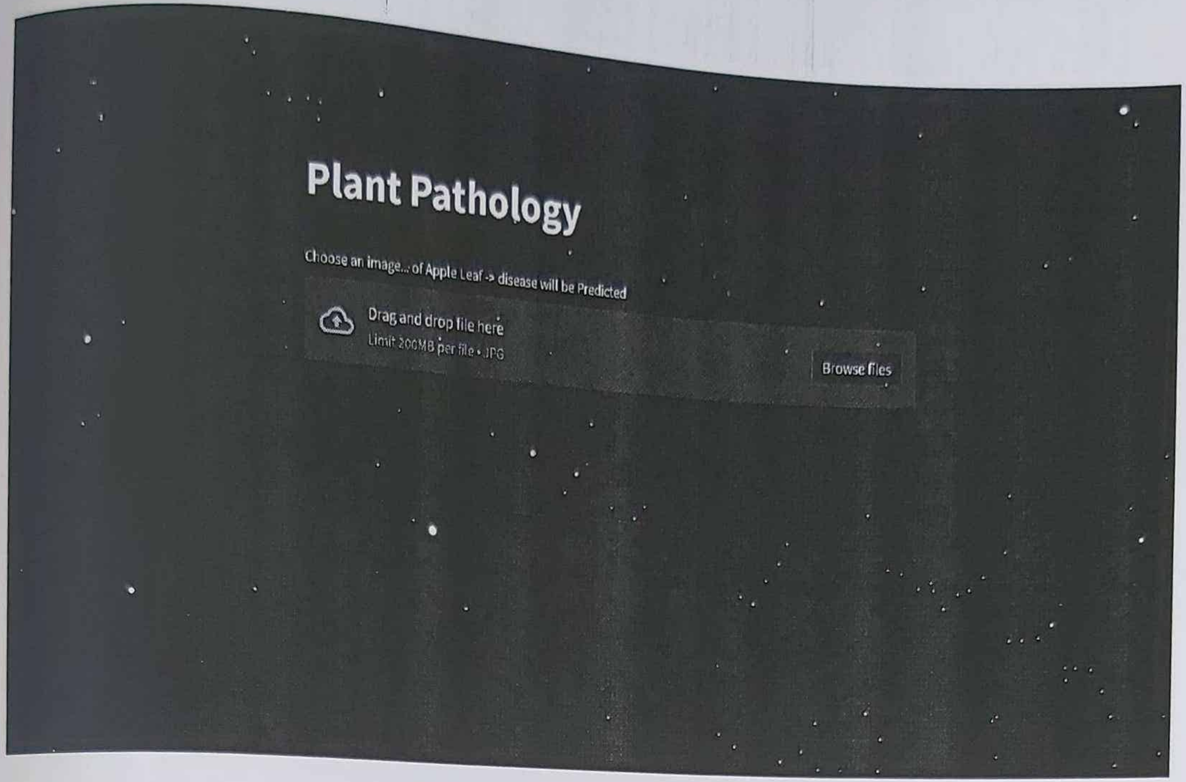


12 Loss

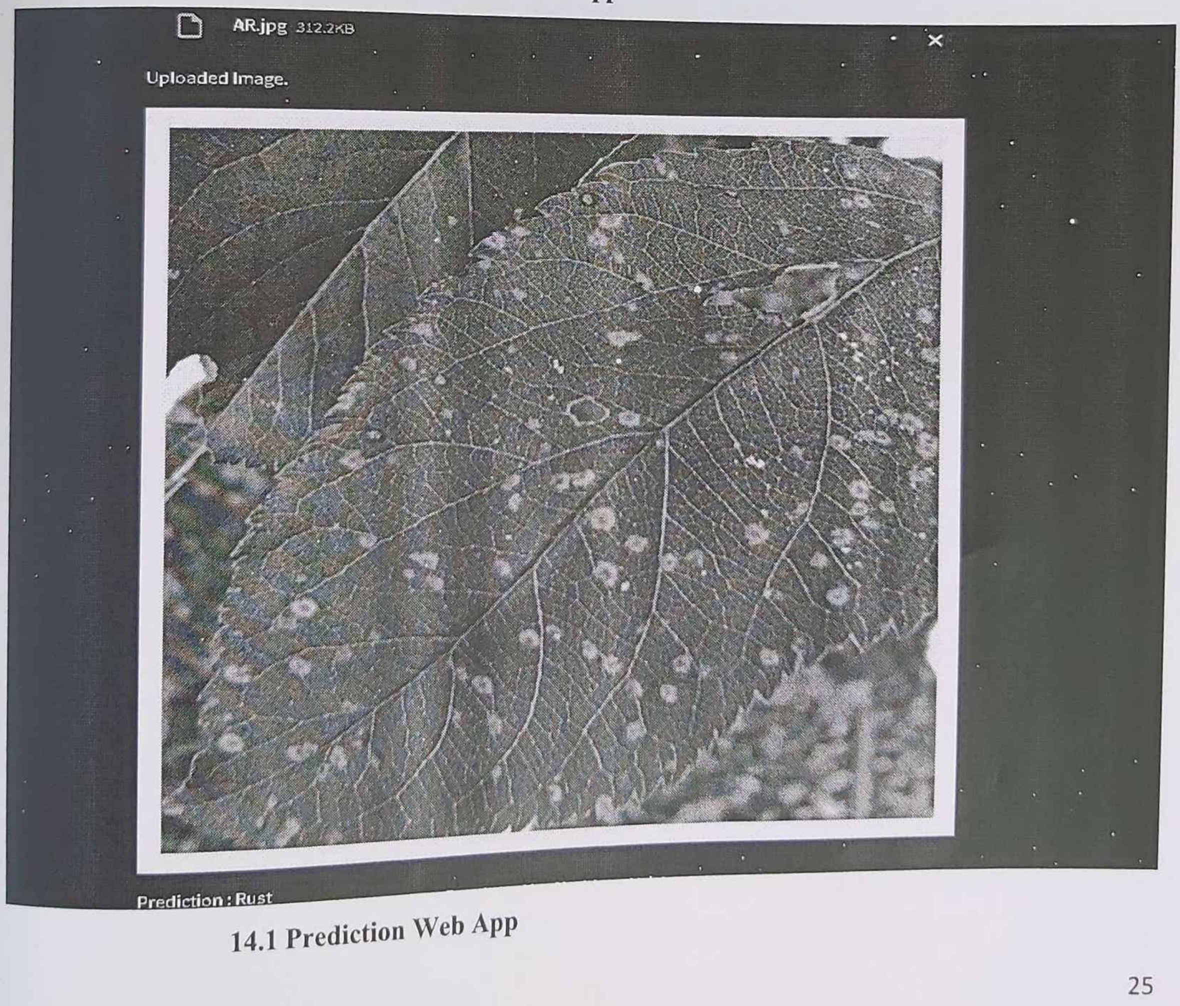
4.2 Result Analysis

There are many methods for automatically finding the disease of plant through Image processing, computer vision and segmentation methods by giving the leaf image as input ,but still ,there are some drawbacks in this area. In this paper, for automatically finding the diseases of plants from images of leaves, we proposed an approach of convolution neural networks.
To analyse the performance of the model, the last result is achieved using parameters such as K-fold cross-validation using 10 folds. RGB-coloured image dataset with augmentation provides 15% best performance for the model. The researchers used the transferred learning CNN model and the grayscale dataset achieved 92.76% accuracy.

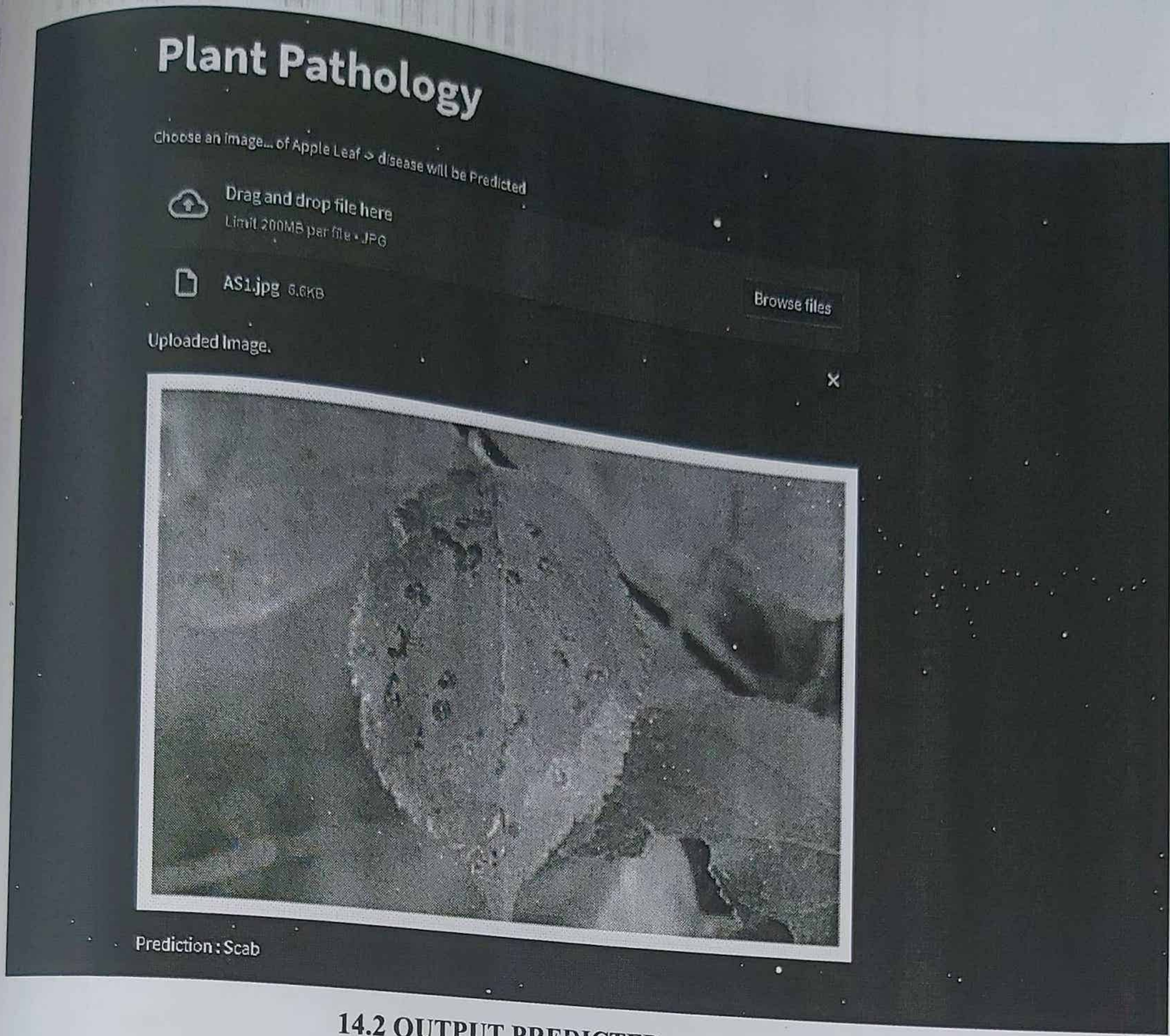
However, colour is the main and most decisive feature in cotton detection and classification; therefore, using a coloured dataset takes a long time to train the model to add performance even if it is a complex layer. Additionally, the researcher has used different pre-processing techniques for noise removal. The main factors for the misclassification of the result exist between bacterial blight, healthy, and leaf miner. The overall performance of the model, as shown in the confusion matrix, is 96.4% accurate for diagnosis of leaf disease and pests of cotton plants.



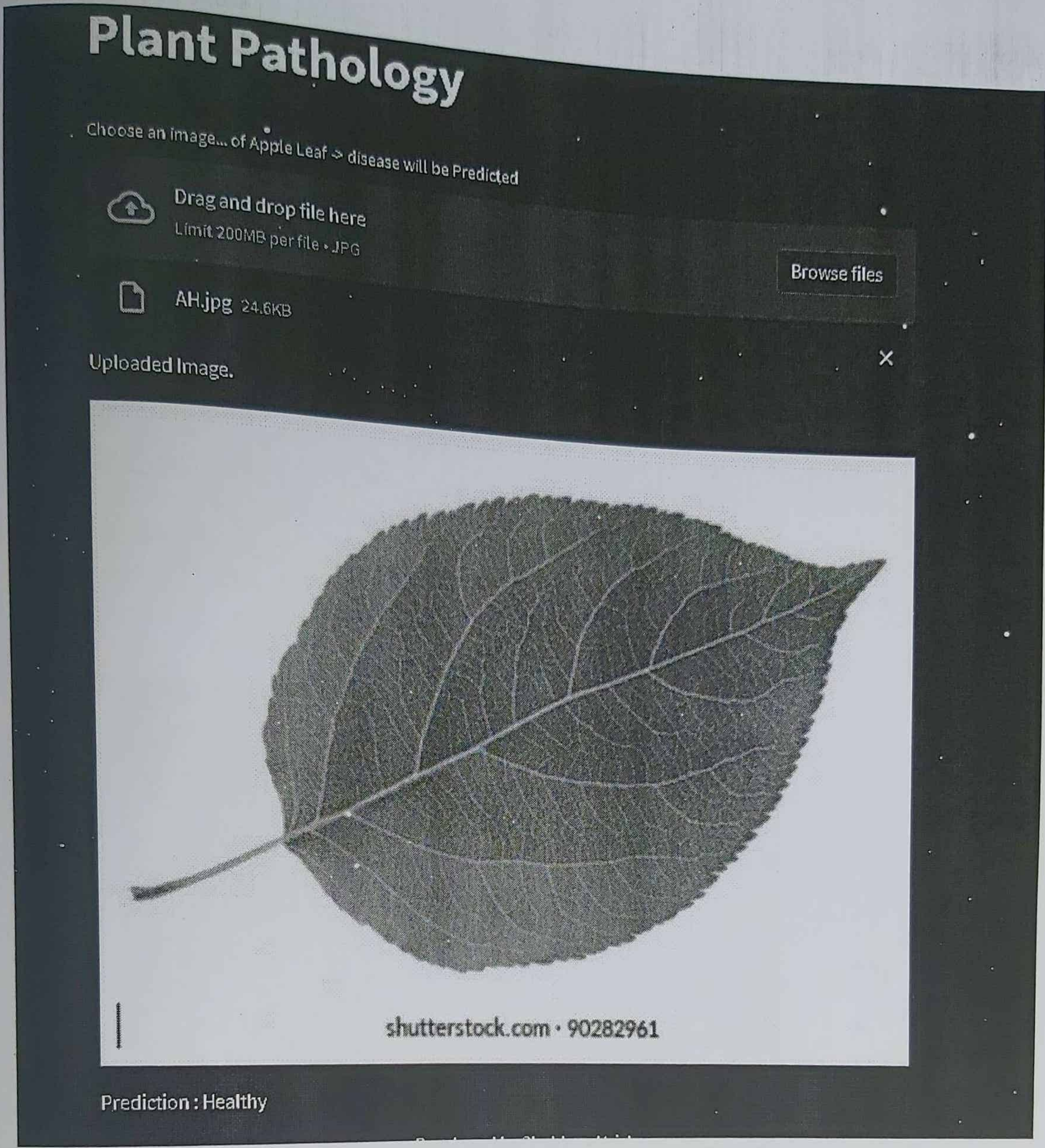
13 Output Screen of Web app



14.1 Prediction Web App



14.2 OUTPUT PREDICTED

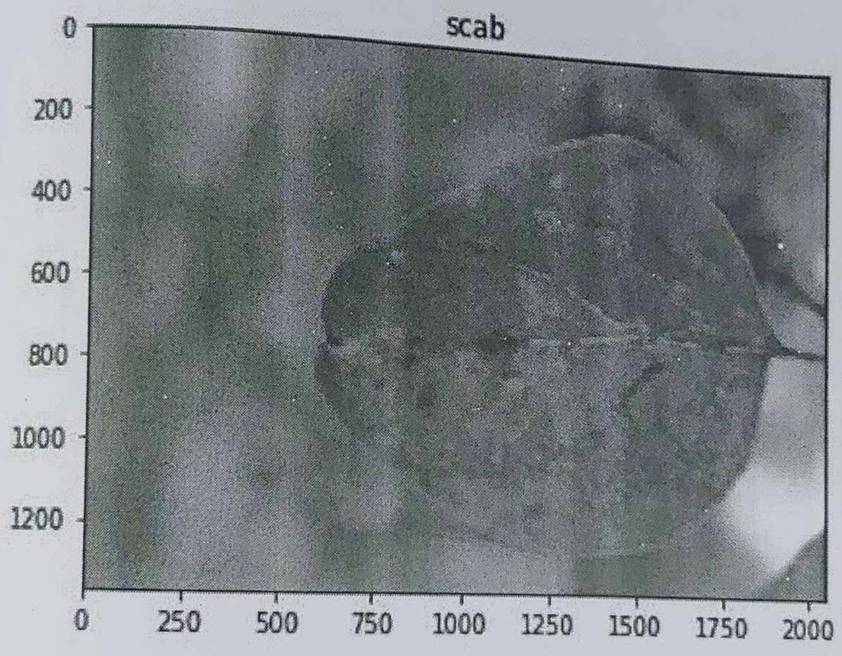


14.3 Output Predicted

```
test_image=r'images\test\Test_7.jpg'
image_result=Image.open(test_image)

from tensorflow.keras.preprocessing import image
test_image=image.load_img(test_image,target_size=(224,224))
test_image=image.img_to_array(test_image)
test_image=test_image/255
test_image=np.expand_dims(test_image,axis=0)
result=model.predict(test_image)
print(np.argmax(result))
Categories=['healthy','multiple_disease','rust','scab']
image_result=plt.imshow(image_result)
plt.title(Categories[np.argmax(result)])
plt.show()
```

3

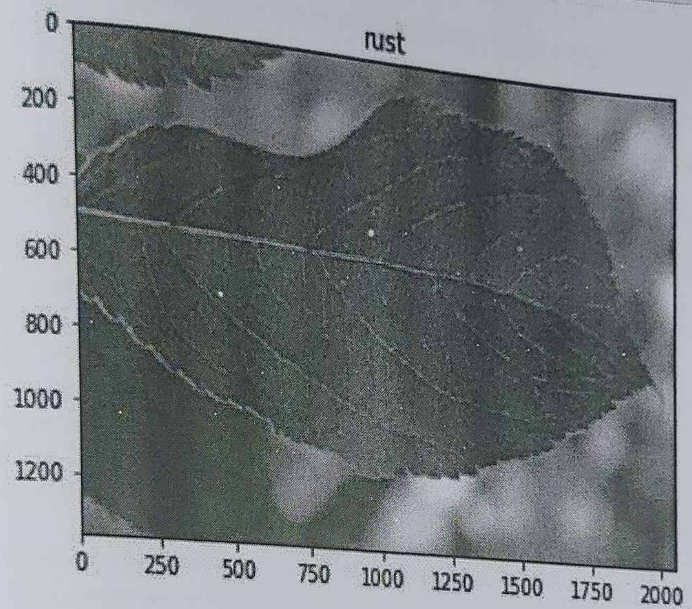


15 Prediction in Jupyter notebook

```
In [44]: test_image=r'images\test\Test_10.jpg'
         image_result=Image.open(test_image)

         from tensorflow.keras.preprocessing import image
         test_image=image.load_img(test_image,target_size=(224,224))
         test_image=image.img_to_array(test_image)
         test_image=test_image/255
         test_image=np.expand_dims(test_image,axis=0)
         result=model.predict(test_image)
         print(np.argmax(result))
         Categories=['healthy','multiple_disease','rust','scab']
         image_result=plt.imshow(image_result)
         plt.title(Categories[np.argmax(result)])
         plt.show()

         2
```



15.2 Rust Prediction

4.3 Application

The overall performance of the cotton leaf disease and pest identification prototype evaluation selected by the evaluator was 60% extremely satisfied option for all questions and 20% of very satisfied and somewhat satisfied option. Also, for the open-ended question, almost all experts reflect constrictive thoughts on the overall performance of the system and prototype. So, this result shows that the prototype of cotton leaf diseases and pests was performed well in problem-solving ability and making a correct prediction is shown in Figure

4.4Problems faced

The main challenge while developing an object detection model on deep learning was to collect a large number of training high-quality images with different shapes, sizes, different backgrounds, light intensity, and orientations in different classes. Therefore, future researchers should try to include a solution for such challenges in their work and not only identify but also suggest remedies for diseases and pests. Installing Libraries like CMAKE OPENCV were difficult due to errors and due to CPU power it took 3-4 hours to train models.

```
91/91 [==============================] - ETA: 0s - loss: 0.2307 - accuracy: 0.9209
Epoch 00026: val_loss improved from 0.25703 to 0.24033, saving model to models\apple3.h5
91/91 [==============================] - 119s 1s/step - loss: 0.2307 - accuracy: 0.9209 - val_loss: 0.2403 - val_accuracy: 0.9091
Epoch 27/30
91/91 [==============================] - ETA: 0s - loss: 0.2350 - accuracy: 0.9216
Epoch 00027: val_loss did not improve from 0.24033
91/91 [==============================] - 118s 1s/step - loss: 0.2350 - accuracy: 0.9216 - val_loss: 0.2805 - val_accuracy: 0.8892
Epoch 28/30
91/91 [==============================] - ETA: 0s - loss: 0.2242 - accuracy: 0.9230
Epoch 00028: val_loss did not improve from 0.24033
91/91 [==============================] - 128s 1s/step - loss: 0.2242 - accuracy: 0.9230 - val_loss: 0.2711 - val_accuracy: 0.9176
Epoch 29/30
91/91 [==============================] - ETA: 0s - loss: 0.2069 - accuracy: 0.9341
Epoch 00029: val_loss did not improve from 0.24033
91/91 [==============================] - 135s 1s/step - loss: 0.2069 - accuracy: 0.9341 - val_loss: 0.2613 - val_accuracy: 0.9006
Epoch 30/30
91/91 [==============================] - ETA: 0s - loss: 0.2172 - accuracy: 0.9223
Epoch 00030: val_loss did not improve from 0.24033
91/91 [==============================] - 141s 2s/step - loss: 0.2172 - accuracy: 0.9223 - val_loss: 0.2560 - val_accuracy: 0.9176
```

4.5 Limitations

Limited to Apple Leaf but Can Predict other Leaf of same disease.

5 Conclusion

There are many methods for automatically finding the disease of plant through Image processing, computer vision and segmentation methods by giving the leaf image as input, but still, there are some drawbacks in this area. In this paper, for automatically finding the diseases of plants from images of leaves, we proposed an approach of convolution neural networks

The complete procedure was explained clearly, from image gathering for training the whole data and validate the trained data to image pre-processing for cleaning the data and features extraction and at last trained deep CNN model. To find the accuracy level of our newly proposed model we performed many tests.

References

S.B. Patil and S.K. Bodhe, "Leaf disease severity measurement using image processing", International Journal of Engineering and Technology, vol.3, no.5, pp.297-301,2011.

2. J.K. Patil and R. Kumar, "Feature extraction of diseased leaf images," Journal of Signal & Image Processing, vol.3, no.1, p.60,2012.

3. R.N. Kadu, S. Kangane, S. Vikhe3, R. Pandita, V. Inamke," Leaf Disease Detection Using Arm7 and Image Processing", Int. Journal of Engineering Research and Applications.ISSN:2248-9622, Vol.5, Issue 2

4. https://www.kaggle.com/c/plant-pathology-2020-fgvc7/overview

5.Apple. Wikipedia. 2021. Available online: https://en.wikipedia.org/wiki/Apple (accessed on 22 April 2021).

6.Jordan, M.I.; Mitchell, T.M. Machine Learning: Trends, Perspectives, and Prospects. *Science* 2015, *349*, 255–26.

7.Badage, A. Crop disease detection using machine learning: Indian agriculture. *Int. Res. J. Eng. Technol.* 2018, *5*, 866–869.

6 Appendices

https://www.sciencedirect.com/science/article/pii/S2214317316300154