AN APPROACH FOR EGGPLANT DISEASE RECOGNITION

BY

Izazul Haque Saad
ID: 181-15-10621

Md. Mazharul Islam
ID: 181-15-10684

AND

Isa Khan Himel
ID: 181-15-10948

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Md. Jueal Mia
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised By

Shah Md. Tanvir Siddiquee
Assistant Professor
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

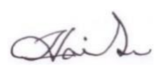
DHAKA, BANGLADESH

JANUARY 2022

APPROVAL

This thesis titled "**AN APPROACH FOR EGGPLANT DISEASE RECOGNITION**", submitted by Izazul Haque Saad, ID No: 181-15-10621, Md. Mazharul Islam, ID No: 181-15-10648, Isa Khan Himel, ID No: 181-15-10948 to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 5th January 2022.

<u>BOARD OF EXAMINERS</u>

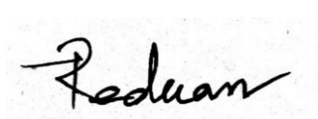


Chairman

Dr. Sheak Rashed Haider Noori (SRH)
Associate Professor and Associate Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Md. Tarek Habib (MTH)
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University



Internal Examiner

Md. Reduanul Haque (MRH)
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

External Examiner

Dr. Mohammad Shorif Uddin
Professor
Department of Computer Science and Engineering
Jahangirnagar University

DECLARATION

We hereby declare that, this thesis has been done by us under the supervision of **Md. Jueal Mia, Senior Lecturer, Department of CSE** Daffodil International University. We also declare that neither this thesis nor any part of this thesis has been submitted elsewhere for award of any degree or diploma.

Supervised by:

6.01.22

Md. Jueal Mia
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised by:

Shah Md. Tanvir Siddiquee
Assistant Professor
Department of CSE
Daffodil International University

Submitted by:

Izazul Haque Saad
ID: 181-15-10621
Department of CSE
Daffodil International University

Md. Mazharul Islam
ID: 181-15-10648
Department of CSE
Daffodil International University

Himel

Isa Khan Himel
ID: 181-15-10948
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Md. Jueal Mia, Senior Lecturer**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "*Computer Vision*" to carry out this thesis. His endless patience ,scholarly guidance ,continual encouragement , constant and energetic supervision, constructive criticism , valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this thesis.

We would like to express our heartiest gratitude to Prof. Dr. Touhid Bhuiyan, Professor, and Head, Department of CSE, for his kind help to finish our thesis and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

Bangladesh's agricultural sector employs the most people. Because Bangladesh is an agricultural country, agricultural goods are the main source of income for the majority of Bangladeshis. The biggest barriers to the growth of our country's agricultural industry are a lack of opportunities and infrastructure, natural calamities, and crop diseases. For farmers of broadacre crops, plant diseases constitute serious productivity and quality limitation. Because significant crops are threatened by a variety of plant diseases and pests, the disease may have an impact on the crop's quality. As a result, it's critical for the farmer to discover the infection at the right time. Crop disease can be effectively monitored and controlled for agricultural and food safety when it can be simply traced out. Because eggplant is a much-needed plant in our nation, we focused our effort on disease detection. If the eggplant is harmed, the country's economy will be severely harmed. Because eggplant has so many properties, it will have a large negative impact on nutrition if it is transited. To identify crop illness quickly and correctly, a technology called computer vision and deep learning is utilized, and we used it to detect eggplant infections. This technique makes it very lucrative and simple to identify plant diseases. This is due to the fact that it decreases the massive workload of crop monitoring and may identify disease symptoms at an early stage. We employed transfer learning models and were able to reach an average accuracy of 99.06% from DenseNet201, followed by Xception's average accuracy of 99.04% and ResNet152V2's average accuracy of 98.93%. We'll achieve this aim by keeping the public in mind and assisting farmers with efficient crop cultivation.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 Introduction

Bangladesh is primarily an agricultural country. Varieties of crops grow abundantly in this country, thanks to its excellent soil and pleasant climate. In FY 2019-20, the agriculture sector contributes around 13.02 percent to the country's GDP and employs around 40.60 percent of the entire labor force [1]. Agriculture is a means of livelihood for 84 percent of Bangladesh's rural people, who rely on it either directly or indirectly [2]. Agriculture employs about half of Bangladesh's population, with more than 70% of the country's territory dedicated to crop production. Rice, jute, wheat, tea, legumes, oilseeds, vegetables, and fruits are among the major crops grown [3]. As a result, providing the long-term security of food for the people of Bangladesh - a densely populated country - requires a sustainable, valuable, and environmentally friendly agricultural system. As a result, the current government has placed the agricultural sector at the top of its priority list in order to ensure Bangladesh's food security. Agriculture production is falling, and its contribution to overall national GDP is falling even faster. The high rate of population expansion, pressure on agricultural land, and climate change are all factors contributing to Bangladesh's falling agricultural growth. A decline in agricultural productivity and a rise in food insecurity have a positive relationship. Bangladesh had self-sufficiency in food production in FY 1999-2000 as compared to its population proportion. Agricultural productivity is dropping year after year as a result of climate change. As a result, it limits the amount of food available in the market to meet market demand. Since the last decade, the government has been importing food from other countries to meet demand. The government of Bangladesh is currently facing significant difficulty in securing food for all [4].

Eggplant is the third most important vegetable in Bangladesh in terms of production, with about 50,000 hectares under cultivation [5]. And with a production of 341000 tons (BBS, 2010), providing for around 25.4 % of the country's total vegetable area. It is grown all over the area in Bangladesh, in all climates and seasons [6]. It has numerous health benefits, including better heart health, digestion, cancer prevention, bone health, anemia prevention, and increased brain function [7]. Antioxidants in eggplant, such as vitamins A and C, help protect our cells from damage. It's also rich in polyphenols, which are natural plant chemicals that may help diabetic cells process sugar more efficiently [8]. The physical method of identifying these diseases is slow and difficult. However, using an automated approach is simple and accurate. As a result, we are using an automated method to assist farmers in identifying diseases before their cultivation, which saves time and money.

1.2 Motivation

Bangladesh is a predominantly agricultural country. As a result, agriculture plays a significant role in our economy. In Bangladesh, eggplants account for a significant percentage of the year's cultivation and production. All across the world, eggplant is a popular vegetable. Regardless of the fact that eggplant is a seasonal crop, it is in high demand in our country. Eggplant, like any other plant, can be infected with a variety of diseases. Our proposed method will not only solve this problem but also assist farmers in increasing eggplant production by identifying diseases early.

1.3 Objective

Farmers suffer financially due to eggplant disease, which has an economic impact on the country. So, in order to reduce production losses, we decided to work on this issue. Our objective is to:

I. Determine the techniques of image processing to classify eggplant diseases.
II. To identify the disease which is causing problems to the farmers in farming.
III. By identifying the disease, farmers can take the necessary steps to control insects, pests, and diseases.
IV. Increase production rate.
V. Save time and deliver standard initial support.

1.4 Features

I. Captured images will be saved in the trained system.
II. Different algorithms will be used to analyze the stored images.
III. A multi SVM classifier is used to categorize the images.
IV. The results of the tests will be provided to the user (farmer) via smartphone.

1.5 Research Question

I. Is it possible for a machine to detect diseases before a human eye does?
II. How can we tell if it's fresh eggplant or not?
III. How can we tell which disease is affecting the eggplant?
IV. What kind of data should be utilized for disease detection training?

1.6 Expected Outcome

Anyone can spot the difference between fresh and infected eggplant. As a result, farmers will be able to take the necessary action. Sellers will be able to choose the right grain, avoiding them from buying rotten grain. Farmers can collect grain samples from their fields to see which diseases have spread. It will be easier for them to produce more grain if they have full knowledge of all of the diseases that affect eggplant. So, they will be able to take the necessary steps to control the diseases.

CHAPTER 2
BACKGROUND

2.1 Introduction

Eggplant disease is a natural occurrence. Ours is an agricultural country, with agriculture being the sole source of income for the vast majority of the population. As a result, agriculture has played a significant part in Bangladesh's overall economic growth. Around 50 % of people in our nation are employed in the agriculture sector. Despite the fact that eggplant is produced all year, it is a seasonal crop that peaks from July to October. It is one of Bangladesh's most important economic sources, and this crop comes in a variety of quality levels. Diseases may attack any plant, and eggplant is no exception. However, it leads to food insecurity. It is also a significant concern for our economy. Many difficulties may be solved by using computer vision to identify eggplant disease. As a result, we believe that this initiative will have a major impact on our agricultural sector. Broadside, it saves farmers time in detecting diseases.

2.2 Literature Review

KR Aravind et al. (2019) [9] were proposed to classify eggplant diseases using AlexNet and VGG-16. They classify five diseases and healthy plants with images acquired from smartphones. They achieved a 93.33% accuracy using the modified VGG-16 model.

In another research KR Aravind et al. (2020) [10] classified eggplant diseases using a Pre-trained model. They pre-trained VGG16 as the feature extractor and MSVM was used to classify eggplant diseases. They used different color spaces (RGB, HVS, YCbCr, and grayscale) to evaluate performance and get the highest classification accuracy of 99.4% using RGB images.

C Xie et al. (2016) [11] research about early blight disease of eggplant leaves with the spectrum and texture features. They classified models using KNN (K-Nearest Neighbor) and AdaBoost and got 88.46% accuracy. In their work, they get hyperspectral images by covering wavelengths. Then, based on the effective wavelengths, they identified gray images. Then they used GLCM (gray level co-occurrence matrix) to extract texture features from hyperspectral and gray images.

Sabrol, H., and Kumar, S. (2019) [12] they have described different types of diseases of tomato and eggplant. In this study they have used neuro-fuzzy classifier for classification. This classifier combines fuzzy logic and neural networks. They used pure grayscale image for analysis and achieved overall accuracy 90.7% for tomato and 98% for brinjal which is quite excellent.

R. Anand et al. (2016) [13] research about image processing technique with k-means clustering method for identifying Brinjal leaves disease. To improve image quality before clustering, the authors used histogram equalization. To extract the colors and texture features, the Color Co-occurrence Method (CCM method) was implemented. The features have been trained using the k-means clustering technique, using three clusters: infected object, infected leaf and the black background of leaf.

To detect eggplant disease Jake (2020) [14] used Image Processing techniques. He used feature extraction to get the prominent features of diseased leaves and fruits. Individual pixels of a color image were broken into RGB values. He performed the MobileNetV2 model to classify different diseases in leaves and fruits.

Wu et al. (2008) [15] analyzed the reflectance intensities of healthy and Botrytis Cinerea on eggplant leaves using a hyperspectral visible near-infrared (VNIR) spectroradiometer. By doing a principal component analysis on the collected data and training a back-propagating NN (Neural Network) using this data, the authors were able to detect fungal disease with 85% before symptoms occurred. The results were based on data collected in a controlled environment with limited ambient lighting. While such VNIR techniques may provide better early detection, they are more complex and expensive to use in automated imaging applications. Spectral angle mapping (SAM), which compares reference and observed spectra by calculating their angular differences when treated as n-dimensional vectors, has also been used to detect disease spread in plants with good success.

Mia et al. (2020) [16] propose a system for In-Depth Exploration of Automated Jackfruit Disease Recognition. They used a k-means clustering segmentation algorithm to split the disease-infected part to extract the features. After employing nine noteworthy classifiers with 480 images of jackfruits, where random forest secured the best position exhibiting 89.59% accuracy.

For cucumber disease identification, Mia et al. (2021) [17] introduced two alternative approaches: traditional machine learning (ML) and computer neural networks (CNN) based transfer learning. They applied k-means clustering to split the image by color and label each pixel, and they used gray level co-occurrence matrix (GLCM) to extract texture and statistical features for disease identification. Using a random forest classifier in machine learning, they were able to achieve an accuracy of 89.93%. They also proposed utilizing InceptionV3, MobileNetV2, and VGG16 to classify cucumber disease, with results of 89.69%, 93.23%, and 90.75%, respectively.

M. Rahman et al. (2021) [32] proposed using CNN to identify pigeon breeds. The authors used a baseline model to test and analyze classification performance. The baseline model performed well despite having just four convolutional layers in its structure, with the highest accuracy of 96.19% during validation and 95.33% during testing.

Mia et al. (2021) [33] has classified five categories of herbs Mehndi, Betel, Mint, Basil, and Aloe Vera. In this paper, the authors have used a neural networks model using YOLO for herb leaves classification and they got 95% accuracy which is impressive when compared to other similar studies.

Maria S.K. et al. (2021) [34] has proposed many techniques for identifying diseases that affect cauliflower plants. The authors compare traditional machine learning with transfer learning in this paper. A segmentation called k-means clustering is utilized to find disease-affected regions of cauliflower pictures, as well as ten relevant features, which are extracted. For classification, they use many classification techniques and they get overall 81.68% accuracy from the Random Forest algorithm. They have also used different types of CNN-based models with transfer learning and they got 90.08% accuracy from InceptionV3 which is the highest accuracy among these two approaches.

Parul et al. (2019) [18] explored potential approaches for automated disease detection in plants using image segmentation. The authors analyzed two models, F-CNN and S-CNN for independent data. To do this, CNN models were trained using segmented visual data. The F-CNN and S-CNN models were compared on the basis of accuracy and data. When compared to the F-CNN model and S-CNN model, the S-CNN model gives the best accuracy of 98.6% when trained on segmental images.

P. Revathi et al. (2013) [19] presented an automated system to detect cotton leaf spot diseases. Cotton leaf images are taken and stored in a database as image features in this approach. The authors used a skew divergence method to extract features such as color, shape and texture. Then select an efficient feature selection method namely Particle swarm optimization (PSO) to select the extracted feature. In this work they use the Cross Information Gain Deep forward Neural Network (CIGDFNN) classifier to identify diseases. This method helps to identify the infected leaf spot of cotton and increases the system's accuracy with a less error rate.

M. Jafari et al. (2017) [20] describe the ability of thermal imaging for pre-symptomatic rose powdery mildew and gray-mold diseases. Selection of superlative thermal features with linguistic hedge values was done for categorizing healthy and infected plants. They use two neuro-fuzzy classifiers for classification. This work showed that the best prediction rates were 69% and 80% (on the second day after inoculation) for identification of pre-symptomatic appearance detection of powdery mildew and gray mold diseases.

R. Zhou et al. (2014) [21] presented an automated system to detect Cercospora Leaf spot in sugar beet. The author has proposed disease detection by robust template matching technique. They used orientation code matching (OCM) method for disease detection. A two-dimensional color histogram and a support vector machine (SVM) approach for pixel-wise disease classification have been used to achieve an accuracy of 87%.

2.3 Comparative Study

Reference	Plant Disease	Method	Average Accuracy	Publication Year
This study	Eggplant	DenseNet201	99.06%	N/A
KR Aravind et al.	Eggplant	VGG-16 (Modified)	93.33%	2019
KR Aravind et al.	Eggplant	VGG16, MSVM	99.4%	2020
C Xie et al.	Early blight disease of eggplant leaves	KNN, AdaBoost	88.46%	2016
Sabrol, H., and Kumar, S.	Tomato and Eggplant	GLCM matrix as feature extractor, ANFIS as classification	90.7%, 98%	2019
R. Anand et al.	Brinjal leaves	K-means clustering	N/A	2016
Jake	Eggplant	MobileNetV2	N/A	2020
Wu et al.	Eggplant Leaves	MSC and SG pretreatment method, Principal component analysis (PCA).	85%	2008
Mia et al.	Jackfruit	K-means clustering, GLCM, Random Forest classifier	89.59%	2020
Mia et al.	Cucumber	GLCM, MobileNetV2	89.93%, 93.23%	2021
Parul et al.	Various types of Plant	F-CNN, S-CNN	98.6%	2019
P. Revathi et al.	Cotton Leaf	Skew divergence method as feature extractor, PSO for feature selection and CIGDFNN for classification	95%	2013
M. Jafari et al.	Rose	Neuro-Fuzzy classifiers	69%, 80%	2017

R. Zhou et al.	Cercospora Leaf spot in sugar beet	OCM, SVM	87%	2014
M. Rahman et al.	Pigeon breeds	CNN baseline model	95.33%	2021
Mia et al.	Herb leaves	YOLO	95%	2021
Maria S.K. et al.	Cauliflower	Random forest, InceptionV3	81.68%, 90.08%	2021

Table 2.3.1: Comparison of our work and other's work

CHAPTER 3

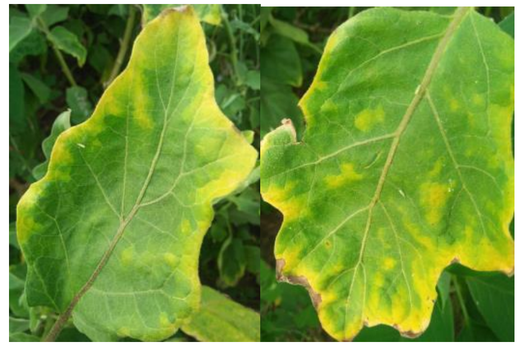
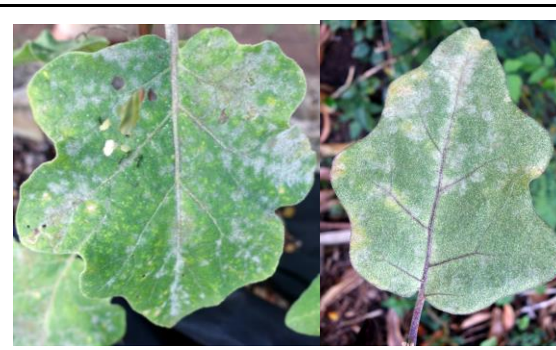
RESEARCH METHODOLOGY

3.1 Data Collection:

One of the main goals of our whole research was to collect our required data as we know for greater accuracy the data collection part is most important. The best-desired data collection makes the most effective accuracy. We collected raw images of eggplant diseases called Aphids, Eggplant fruit and shoot borer, Cercospora, Flea Beetle, Fruit rot, Leaf curl, Leaf roller, Leafhopper, Mealybug, Powdery mildew, Spider mite, Spotted beetle, Thrips and Whitefly as well as healthy eggplants. We captured those raw images from Sher-e-Bangla Agricultural University and Bangladesh Agricultural Research Institute's field. We collected these data between 2020-2021. We also collected some data from internet sources for increasing the accuracy levels.

Class Name	Train	Test	Total Image	Samples
Aphids	79	20	99	
Eggplant Shoot And Fruit Borer	202	51	253	

Cercospora	258	64	322	
Flea Beetle	125	32	157	
Fruit Rot	89	23	112	
Leaf curl	117	30	147	
Leaf Roller	29	8	37	

Leafhopper	84	21	105	
Mealybug	244	61	305	
Healthy Fruit	211	53	264	
Healthy Leaf	199	50	249	
Powdery Mildew	19	5	24	

Spider Mite	240	61	301	
Spotted Beetle	187	47	234	
Thrips	80	21	101	
Whitefly	44	12	56	

Table 3.1.1: Data Samples

3.2 Description of Eggplant Disease:

1. **Aphids:**

   In most cases, low to moderate quantities aren't harmful to crops. A severe infestation can cause leaves and shoots to curl, wilt, or yellow, limiting the plant's growth. There will be an overall fall in plant growth. Aphids create honeydew, which can lead to secondary infection by opportunistic fungi, as seen by mold growth on the leaves. Ants are drawn to honeydew. Even modest numbers of aphids can spread viruses from plant to plant over time.

Fig. 3.2.1: Aphids diseases

2. **Eggplant fruit and shoot borer:**

   Brinjal fruit and shoot borer (BFSB) is a serious pest of brinjal and one of the most significant obstacles to brinjal production. It's an internal borer that eats the tender shoots and fruits of the plant. The larvae of the moth, Leucinodes orbonalis, are responsible for the damage. The first noticeable sign is the withering of branch tips due to early larval feeding. Flowers, flower buds, and stems are all damaged later on. It's possible that the fruits produced by those plants aren't safe and suitable for eating. When a large population has accumulated over multiple generations, the damage is the most severe.

Fig. 3.2.2: Eggplant fruit and shoot borer diseases

3. **Cercospora:**

   Cercospora melongenae is an eggplant fungus that may live in infected plant waste or soil for at least a year. Disease growth is aided by wet leaves and high relative humidity. The infection can appear on leaves, petioles, and stems at any stage of growth. Eggplant fruits are not harmed, however severe foliar diseases can limit production. On the top surface of the older, lower leaves, the first symptoms show as tiny, round, and slightly sunken spots. The specks get bigger, more irregular, and encircled by a yellow halo as time passes. Later on, the leaf dots may be seen on both sides of the leaf. Despite the fact that the fungus does not directly infect the fruits, the lower production of the plants might result in lowered fruit growth.

Fig. 3.2.3: Cercospora diseases

4. **Flea Beetle:**

   Flea beetles are a diverse group of insects that attack a wide range of plants. The majority of adults are tiny (approximately 4mm), dark-colored, and have a glittering or metallic appearance. Their body is round, and their rear legs are huge. They have an expanded thigh on their hind legs and can jump and fly long distances. The adults eat leaves. Small dispersed shot-like holes (1-2 mm) and small chewing cavities that do not cut through the leaf blade (pitting) characterize the damage. A minor yellowing of the damaged area is possible.

Fig. 3.2.4: Flea Beetle diseases

5. **Fruit rot:**

Phomopsis vexans, a pathogen that appears to be confined to eggplants, causes the symptoms. The fungus persists in plant waste, and its spores are spread to healthy plants by wind and rain. The first signs of the disease show in the leaves, stems, and fruits, with the latter being the most visible. On leaves, little gray to brown spots with bright centers emerge, which can grow in number and cover significant portions of the leaf blade. Fruits develop brown, squishy, sunken spots. They typically combine as they grow larger, covering a considerable percentage of the fruit's surface and forming concentric rings of little black dots on the fruit's edges. Fruits rot over time.

Fig. 3.2.5: Fruit rot diseases

6. **Leaf curl:**

Spotted wilt virus is one of the most frequent infections that cause leaf curling. It's a virus that attacks eggplant leaves and causes them to wilt.

Fig. 3.2.6: Leaf curl diseases

7. **Leaf roller:**

The eggplant is severely harmed by the Leaf Roller, a pest. The white of the hindwings is transparent. The larvae are the only ones who cause leaf damage. Where the larvae are found, the first signs are lengthwise curled leaves. The inside green tissue of the leaves is then chewed on. The majority of the damage is seen on the plant's upper layers. Browning, wilting, and drying rolled leaves may be damaged. Browning spreads to whole plant parts when the damage is severe, and defoliation follows. If the insect population is not managed, this can result in a large loss of production.

Fig. 3.2.7: Leaf roller diseases

8. **Leafhopper:**

Several species of Amrasca leafhoppers are responsible for the damage. They are primarily greenhouse pests, and even small numbers can create problems. The plant's response to drought stress or nutritional deficits is similar to Leafhopper's Damage. Leafhoppers eat the leaves' undersides. Chlorotic patches surrounding the feeding tips and a slight rolling of the leaf edges are common signs of damage. The symptoms are commonly referred to as "hopper burn." The chlorotic regions eventually spread to the rest of the leaf. Leaves may curl downward before falling off completely. Plant growth is inhibited, and fruit production may be reduced as a result.

Fig. 3.2.8: Leafhopper diseases

9. **Mealybug:**

Crawlers (first-instar nymphs) appear on the underside of leaves on terminal shoots, stems, and other plant components, indicating the initiation of a Mealybug infestation of above-ground plant parts. In warm or temperate areas, mealybugs are oval, wingless insects. On the undersides of leaves, stems, flowers, and fruits, white cotton-like masses made up of flocks of bugs develop.

Fig. 3.2.9: Mealybug diseases

10. Powdery mildew:

It is a fungus that may infect a variety of plants. Many distinct species of fungus in the Erysiphales order cause powdery mildew infections. Powdery mildew is one of the simpler plant diseases to recognize because of its characteristic symptoms. White powdery patches appear on the leaves and stems of infected plants.

Fig. 3.2.10: Powdery mildew diseases

11. Spider mite:

Spider mites belong to the Acari (mite) family Tetranychidae, with over 1,200 different species. They usually exist on the undersides of plant leaves, where they may weave protective silk webs, and they can harm plant cells by puncturing them to eat.

Fig. 3.2.11: Spider mite diseases

12. Spotted beetle:

Larvae and adults of the Colorado potato beetle (Leptinotarsa decemlineata) devour eggplant leaves with their chewing mouthparts. Populations can soon grow out of control, resulting in substantial defoliation and yield loss. Adults and larvae are easy to handpick and squash in small gardens or put in a bottle of soapy water. Chemical management may be required in bigger gardens.

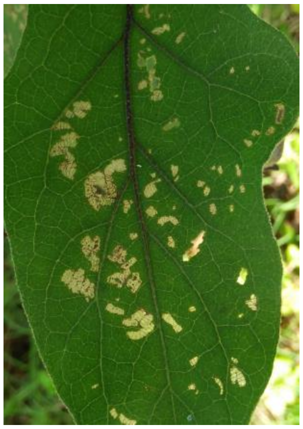


Fig. 3.2.12: Spotted beetle diseases

13. Thrips:

Thrips damage the flowers and fruits of the eggplant, resulting in a decrease in production. Thrips larvae and adults eat and lay eggs on new eggplant leaves, flowers, and fruits.

Fig. 3.2.13: Thrips diseases

14. Whitefly:

Whiteflies wreak havoc on eggplants by taking lots and lots of sap and coating the plants in sticky honeydew. The honeydew is covered with black sooty mold, which reduces the plant's photosynthetic ability and makes the fruit ugly.

Fig. 3.2.14: Whitefly diseases

3.3 Research Methodology:

The most important aspect of every research project is the methodology. A research project can progress step by step and attain its aim with efficiency. It is also feasible to learn new things with the right approaches and plans. We also used various approaches to work on the eggplant disease recognition dataset. Data collection, preprocessing, data augmentation, model training (DenseNet201, Xception, ResNet152V2), and model evaluation are the steps in the approach.

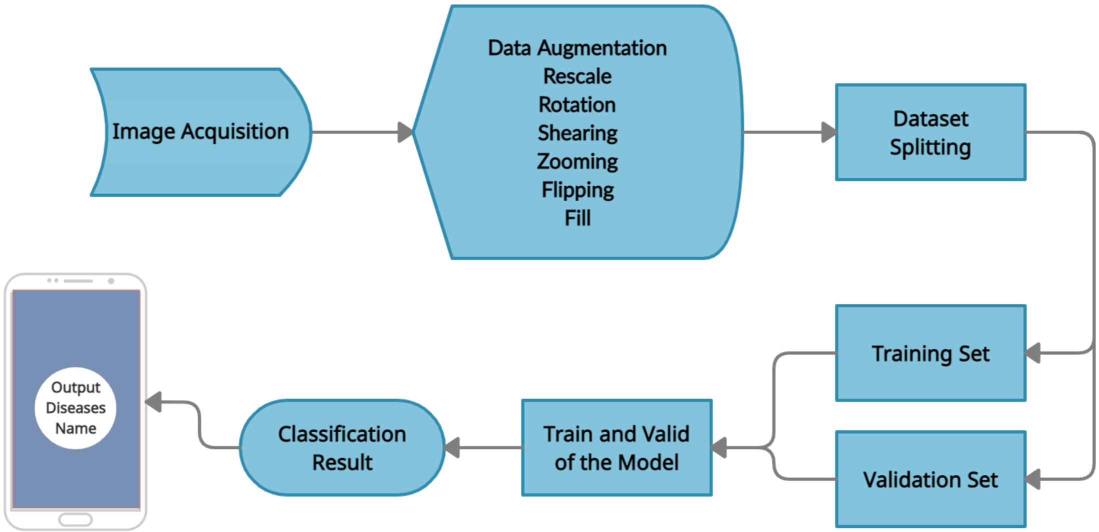


Fig. 3.3.1: Working Process of Eggplant Disease Recognition

3.3.1 Image Acquisition:

The process of obtaining a picture from a source is known as image acquisition. So, this is our first step to collecting data for training and testing purposes. Image data was captured from the field with the help of a mobile phone. So, we have collected data with cameras from the field and some from the Internet. Collected image data contains both healthy and infected leaves and fruits. All our collected data is in RGB format. Here is a sample image of the Brinjal shoot and fruit borer shown in fig 3.3.1.

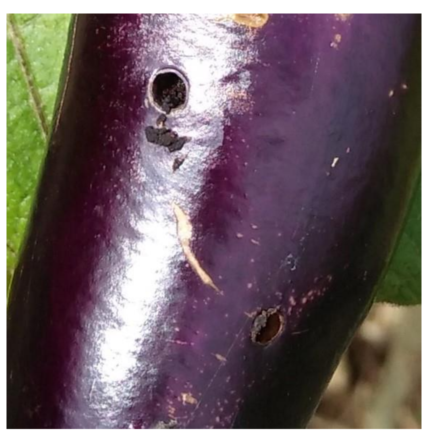


Fig. 3.3.1.1: Infected eggplant

3.3.2 Data Preprocessing:

The actions taken to prepare pictures before they are utilized in model training and inference are referred to as image preprocessing. To clean picture data for model input, preprocessing is necessary. Convolutional neural networks, for example, required that all pictures be the same size arrays [22].

3.3.3 Data Augmentation:

Raw pictures are images that have been taken from multiple sources and cannot be used immediately in the next step. So, we must apply preprocessing in the following step to convert images to an acceptable format in order to improve the image's quality for future processing. Image processing entails eliminating noise or undesired objects, cropping the image to the desired size and form, enhancing and filtering the image, and so on. We have used the following steps:

- **Image resizing:** We have resized the images into different resolutions like 224×224, 299×299.
- **Rescale:** As we know, pictures get divided into pixels, and the color value stays 0-255. The maximum pixel value is 255. Every pixel value from the range [0,255] is transformed to [0,1] using Rescale 1./255.
- **Rotation:** This rotates the picture clockwise by a random amount of degrees within 0 and 360 degrees.
- **Shearing:** The term 'shear' refers to a distortion of an image along an axis, usually to generate or correct perception angles. It's typically used to enhance photographs, so computers are able to see how people perceive things from various perspectives.
- **Zooming:** Zooming in and out of a picture can be done randomly. For example, a zoom range of [0.7, 1.3] shows zooming within 70 percent (zoom in) and 130 percent (zoom out).
- **Flipping:** Horizontal flip is a way for horizontally flipping both rows and columns. Vertical flip means vertically flipping both rows and columns.
- **Fill Mode:** We can fill our image with different modes like Reflect, Wrap, Constant, Nearest.

3.3.4 Train Model with Training Dataset:

We used our training dataset, which has 2207 eggplant images, to train three different CNN models (Convolutional Neural Network). We went with the following model:

- DenseNet201
- Xception
- ResNet152V2

3.3.5 Model Evaluation:

To build an effective convolutional neural network model, model evaluation is essential. For testing our model, we evaluated the confusion matrix, accuracy, F1-score, precision, recall, and plot diagram.

3.4 CNN

CNN is a sort of artificial neural network that is mostly used for image processing and recognition. A CNN is a powerful tool, but it must be trained with millions of labeled data points. If CNN's are to deliver results quickly enough to be helpful, they must be trained with powerful processors, such as a dedicated GPU. While CNN's are developed to address problems with visual vision, they can also be used for image categorization, NLP, drug development, and health risk assessments. Self-driving automobiles can also benefit from CNNs for depth estimation.

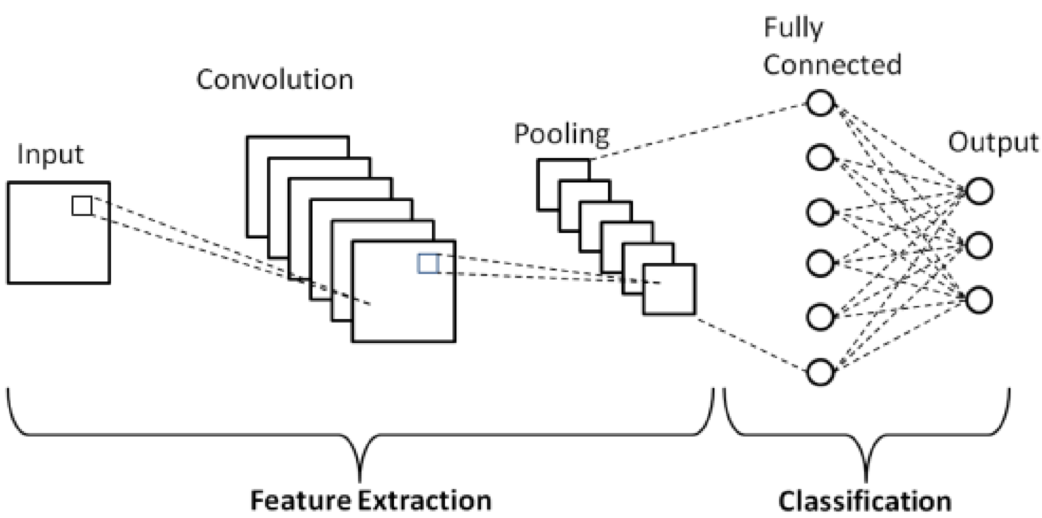


Fig. 3.4.1: CNN Architecture [23]

3.5 Transfer Learning

The reuse of a previously trained model on a new platform is referred to as transfer learning. It's gaining popularity in deep learning since it can train deep neural networks with a small amount of data. This is particularly beneficial in data science, as most real-world situations do not require millions of labeled data points to train these complicated models. Traditional machine learning models take longer to attain optimal performance than transfer learning models. It's because models that employ previously trained models' knowledge (features, weights, and so on) already know what the features are. It is quicker than training neural network models from the ground up.

3.6 Pre-trained Models:

There are many effective pre-trained models for image classification. As we already said, we have used three transfer models for our research. But, we have trained more than five models, and among them, DenseNet201, Xception, and ResNet152V2 performed better.

3.7 DenseNet201:

DenseNet is a feed-forward network design in which each layer is directly linked to every other layer (in each dense block). All previous layers' feature maps are handled as discrete inputs for each layer, while its feature maps are forwarded on as feeds to all future layers. For CIFAR10/100 (with or without data preprocessing) and SVHN, this connection pattern gives state-of-the-art accuracies. DenseNet achieves equivalent accuracy as ResNet on the large-scale ILSVRC 2012 (ImageNet) datasets, but with less than half the number of parameters and nearly half the number of FLOPs [24]. For DenseNet2011, the input image size has to be (224, 244, 3) [25].

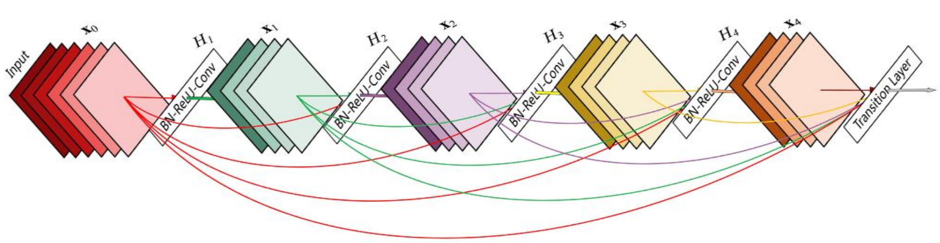


Fig. 3.7.1: A five-layer dense block with a growth rate of 4 [26]

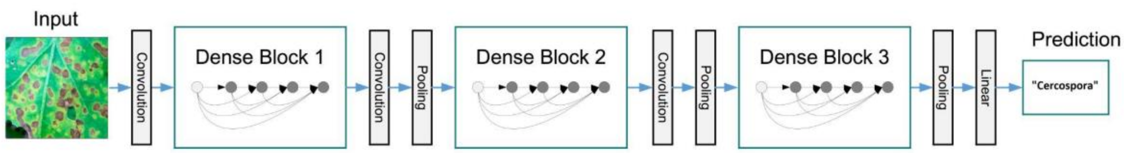


Fig. 3.7.2: DenseNet containing three dense blocks

3.8 Xception:

Xception is a review of the Extreme version of Inception. It's even better than Inception-v3 using a modified depthwise separable convolution. To construct the original depthwise separable convolution, a pointwise convolution is used after the depthwise convolution. The depthwise convolution is a channel-wise n×n spatial convolution. If we have four channels, we will have four n×n spatial convolutions. To change the dimension, pointwise convolution is the 1×1 convolution.

The pointwise convolution is preceded by a depthwise convolution to form the enhanced depthwise separable convolution. This change is inspired by the inception module in Inception-v3's that 11 convolutions be performed before any n×n spatial convolutions. As a result, it varies slightly from the initial. Since convolutions over all channels are no longer required for this modification, the model becomes substantially lighter and has fewer connections. The overall architecture of the Xception model is divided into 3 parts: entry flow, middle flow, and exit flow. Modified depth-wise separable convolutions are considered inception modules in the architecture and appear in all three parts, along with backpropagation [27]. For Xception, the default input image size is (299, 299, 3) [28].

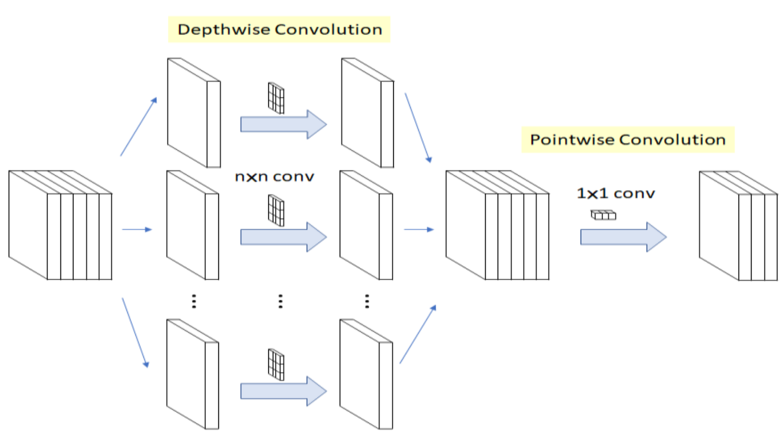


Fig. 3.8.1: Depthwise Separable Convolution

3.9 ResNet152V2:

Residual networks, or ResNet, have such a similar design to the VGG-16, but due to its identity mapping capabilities, ResNet provides a superior solution to the infamous vanishing gradient problem. It took top place in the ImageNet and Coco competition at the ILSVRC and COCO 2015. ResNet invented the identity shortcut connection, which allowed us to successfully train convolutional neural networks with layers of up to 150. The bypass link allows the network to learn an identity function and address the vanishing gradient problem. This function assures that the top layers perform as well as the lower ones. For ResNet152V2, the input image size has to be (224, 244, 3) [29].

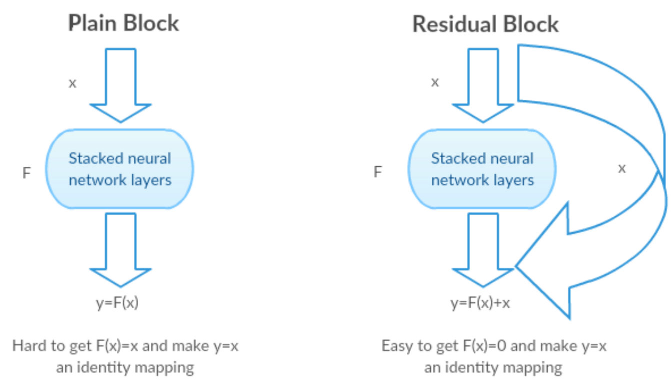


Fig. 3.9.1: Identity Mapping of ResNet [30]

CHAPTER 4

IMPLEMENTATION STEPS

4.1 Introduction

Image classification mainly depends on the pixels. This is tough to find or identify, classify disease and recognize it. These trained models work to look for spots and then classify the image. When it comes to classification, these models play an important role. In this chapter, we will discuss the implementation of our models. As these models' implementation is almost the same, we'll show it with DenseNet201.

4.2 Used Tools

Tools are a must when we implement a project. During our research, we had to use some tools and libraries, which we'll discuss further.

4.2.1 Google Colaboratory

Google Colaboratory is a free online Jupyter notebook environment that lets us train ML and DL models on CPUs, GPUs, and TPUs. We can use GPUs and TPUs for free for a limited time but can be used again after some moment. ML and DL models can be trained faster on GPUs and TPUs.

4.2.2 TensorFlow

TF or TensorFlow is a free of charge and publicly available ML framework. It's used to build DL and ML applications. The Google team designed TF to create and explore interesting AI concepts. Because TF is written in the Python programming language, it is called a user-friendly framework.

4.2.3 Keras

Keras is a Python-based publicly available and free-of-charge neural network API that relies on Theano or Tensorflow. It's built to be flexible, quick, and simple to use. Google developer François Chollet invented it. Low-level computing is not handled by Keras. Instead, it makes use of a library known as the "Backend."

4.3 Implementation procedures

- **Library import:** We need to import some libraries like TensorFlow, Keras, etc., to work with our model on google colaboratory.

```
import tensorflow as tf
import keras
from keras.layers import Input, Lambda, Dense, Flatten
from keras.models import Model
from keras.preprocessing import image
from tf.keras.applications.densenet import DenseNet201
from keras.preprocessing.image import ImageDataGenerator
from keras.models import Sequential
from keras.callbacks import ModelCheckpoint
import numpy as np
from glob import glob
import matplotlib.pyplot as plt
```

Fig. 4.3.1: Importing libraries

- **Image size and dataset import:** We discussed earlier that different models take the different shapes of image input. For our DenseNet201 model, the input image shape is (224, 224, 3). Here, both the height and width of the image are 224. And ‘3’ is for Red, Green, Blue (RGB) format. The batch size we used in this model is the same as other models we implemented, which is 16. And next, we assigned the train and test path in two variables.

```
IMAGE_SIZE = [224, 224]
batch_size = 16

train_path = '/content/drive/MyDrive/Project Dataset/Modified Dataset/Dataset/train'
test_path = '/content/drive/MyDrive/Project Dataset/Modified Dataset/Dataset/test'
```

Fig. 4.3.2: Data import

- **Data augmentation:** With the ‘ImageDataGenerator’ class, we can do data augmentation easily. Our procedure for other models is the same as this. We have used rescale, shear, rotation, zooming, fill mode, flipping all these functions to improve our model performance and output with new and dissimilar datasets. As we have multiclass, our class mode is ‘categorical,’ giving a 2D result which is one-hot encoding. And color mode is ‘RGB as all of our data is in RGB format.

```
train_datagen = ImageDataGenerator(
                                  rescale=1./255,
                                  shear_range=0.2,
                                  rotation_range=20,
                                  zoom_range = 0.2,
                                  fill_mode='reflect',
                                  vertical_flip=True,
                                  horizontal_flip=True)

training_set = train_datagen.flow_from_directory(train_path,
                                                 target_size = (224, 224),
                                                 batch_size = batch_size,
                                                 class_mode = 'categorical',
                                                 color_mode='rgb',
                                                 shuffle=True)

test_datagen = ImageDataGenerator(rescale = 1./255)

test_set = test_datagen.flow_from_directory(test_path,
                                            target_size = (224, 224),
                                            batch_size = batch_size,
                                            color_mode='rgb',
                                            shuffle=True,
                                            class_mode = 'categorical')
```

Fig. 4.3.3: Data augmentation

- **Model build:** First, we call the DenseNet201 model we previously imported and give its image input shape as previously declared. And, the weights are the default values from 'Imagenet.' The relative probabilities are determined by the 'Softmax' activation function.

```
densenet201 = tf.keras.applications.densenet.DenseNet201(input_shape=IMAGE_SIZE + [3], weights='imagenet', include_top=False)

for layer in densenet201.layers:
  layer.trainable = False

x = Flatten()(densenet201.output)
prediction = Dense(len(folders), activation='softmax')(x)
model = Model(inputs=densenet201.input, outputs=prediction)

model.summary()
```

Fig. 4.3.4: Model build

- **Model compile:** The loss function, optimizer, and metrics are all interpreted by compile. We used 'categorical crossentropy' as loss function, 'Adam' as an optimizer, and 'accuracy' for metrics.

```
model.compile(
  loss=keras.losses.categorical_crossentropy,
  optimizer = tf.keras.optimizers.Adam(),
  metrics=['accuracy']
)
```

Fig. 4.3.5: Compiling model

- **Model Training:** In this part, we trained the model with 50 epochs.

```
history = model.fit_generator(
    training_set,
    validation_data=test_set,
    epochs=50,
    steps_per_epoch=len(training_set),
    validation_steps=len(test_set)
)
```

Fig. 4.3.6: The raining model

CHAPTER 5

EXPERIMENTAL RESULTS AND DISCUSSION

5.1 Introduction:

Using five CNN-based transfer learning models, this chapter will demonstrate our training and testing results. For each model, we split the test dataset into sixteen classes, and the results for each are listed below. We used 2766 eggplant images in total, with 80% (2207 images) of data allocated for training and 20% (559 images) reserved for testing. For higher accuracy, we augmented our training and testing data.

Batch: The amount of training samples used in one iteration is referred to as batch size in machine learning.

Epoch: An epoch is a ML term that refers to how many trips the ML algorithm has made across the full training dataset.

Confusion matrix: A confusion matrix is a special form of table design used in ML (Machine Learning) and associated disciplines. It is useful to demonstrate prediction and recall in a system where the values are known for test data.

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Fig. 5.1.1: Confusion matrix [31]

5.2 Experimental Results:

We compared the performance of different types of CNN-based transfer learning approaches using classification model performance metrics. Each of the models created a confusion matrix, which we analyzed. M, or the multiclass confusion matrix, is a n * n square matrix with n rows and n columns and n^2 entries. Because we worked on sixteen classes, each model yields a 16 * 16 confusion matrix. Shows performance evaluation matrices for multiclass confusion matrix. The performance evaluation measures in percentage are calculated using the following formula: accuracy, precision, specificity, sensitivity, FNR, and FPR.

Accuracy = $\left(\frac{TP+TN}{TP+FP+FN+TN} \times 100\right)$ %

Precision = $\left(\frac{TP}{TP+FP} \times 100\right)$ %

Specificity = $\left(\frac{TN}{FP+TN} \times 100\right)$ %

Sensitivity = $\left(\frac{TP}{TP+FN} \times 100\right)$ %

FNR = $\left(\frac{FN}{TP+FN} \times 100\right)$ %

FPR = $\left(\frac{FP}{FP+TN} \times 100\right)$ %

Confusion Matrix:

In Figure 5.2.1, 5.2.2, and 5.2.3, the generated confusion matrix for each of the models is given below. Here, ‘A’ denotes Aphids, ‘B’ denotes Brinjal Shoot And Fruit Borer, ‘C’ denotes Cercospora, ‘D’ denotes Flea Beetle, ‘E’ denotes Fruit Rot, ‘F’ denotes Leaf Curl, ‘G’ denotes Leaf Roller, ‘H’ denotes Leafhopper, ‘I’ denotes Mealybug, ‘J’ denotes Normal Fruit, ‘K’ denotes Normal Leaf, ‘L’ denotes Powdery Mildew, ‘M’ denotes Spider Mite, ‘N’ denotes Spotted Beetle, ‘O’ denotes Thrips, and ‘P’ denotes Whitefly.

Actual

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
A	12	0	0	0	0	0	0	0	2	0	0	0	0	0	2	3
B	0	46	0	0	0	0	0	0	0	2	0	0	0	0	0	0
C	0	0	64	1	0	0	1	0	2	0	0	0	0	1	0	0
D	1	0	0	28	0	0	0	0	0	0	0	0	0	3	0	0
E	0	0	0	0	19	0	0	0	0	4	0	0	0	0	0	0
F	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	30	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	0	0	21	0	0	0	0	0	0	0	0
I	1	0	1	0	0	0	0	0	59	0	0	0	0	0	0	0
J	0	0	0	0	1	0	0	0	0	52	0	0	0	0	0	0
K	0	0	0	1	0	0	0	0	0	0	48	0	0	0	1	0
L	0	0	1	0	0	0	0	0	0	0	0	3	0	1	0	0
M	0	0	0	0	0	0	0	0	0	0	0	0	61	0	0	0
N	0	0	0	0	0	0	0	0	1	0	1	0	0	45	0	0
O	1	0	0	1	0	0	0	0	0	2	0	0	0	0	14	2
P	2	0	0	2	0	0	0	1	0	0	0	0	0	0	0	7

Predicted

Fig. 5.2.1: Confusion Matrix of DenseNet201

Actual

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
A	12	0	1	0	0	0	0	0	1	0	1	0	0	1	0	3
B	0	47	0	0	1	0	0	0	0	0	0	0	0	0	0	0
C	1	0	66	0	0	0	0	0	0	0	0	1	1	0	0	0
D	1	0	0	27	0	0	0	1	1	0	0	0	1	0	0	1
E	0	2	0	0	19	0	0	0	0	2	0	0	0	0	0	0
F	0	0	0	0	0	6	0	0	0	0	0	0	0	0	2	0
G	0	0	0	0	0	0	30	0	0	0	0	0	0	0	0	0
H	0	0	0	0	0	0	0	20	0	0	0	0	0	1	0	0
I	0	0	0	0	0	0	0	1	58	0	0	0	0	1	0	1
J	0	0	0	0	0	0	0	0	0	53	0	0	0	0	0	0
K	0	0	0	0	0	0	0	0	0	0	49	0	1	0	0	0
L	0	0	1	0	0	0	0	0	2	0	0	2	0	0	0	0
M	0	0	0	0	0	0	0	0	0	0	0	0	61	0	0	0
N	0	0	1	0	0	0	0	0	4	0	0	0	0	42	0	0
O	2	1	0	1	1	0	0	0	0	0	0	0	0	0	14	1
P	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10

Predicted

Fig. 5.2.2: Confusion Matrix of Xception

Actual

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
A	15	0	1	1	0	0	0	0	1	0	0	0	0	0	1	0
B	0	47	0	0	1	0	0	0	0	0	0	0	0	0	0	0
C	0	0	69	0	0	0	0	0	0	0	0	0	0	0	0	0
D	1	0	2	27	0	0	0	0	0	0	0	0	0	1	1	0
E	0	3	0	0	17	0	0	0	0	3	0	0	0	0	0	0
F	0	0	0	0	0	8	0	0	0	0	0	0	0	0	0	0
G	0	0	0	0	0	0	29	0	0	0	0	0	0	0	1	0
H	0	0	0	0	0	0	0	21	0	0	0	0	0	0	0	0
I	0	0	1	0	0	2	0	0	52	0	0	0	0	4	1	1
J	0	0	0	0	2	0	0	0	0	51	0	0	0	0	0	0
K	0	0	1	0	0	0	0	0	0	0	49	0	0	0	0	0
L	0	0	4	0	0	0	0	0	0	0	0	1	0	0	0	0
M	0	0	1	0	0	0	0	0	1	0	1	0	57	1	0	0
N	0	0	3	0	0	0	0	0	0	0	1	0	0	43	0	0
O	1	0	0	0	1	0	0	0	0	0	0	0	0	0	17	1
P	2	0	0	1	0	0	0	0	0	0	0	0	0	0	1	8

Predicted

Fig. 5.2.3: Confusion Matrix of ResNet152V2

We have evaluated our three models with a confusion matrix from figure 5.2.1, 5.2.2, and 5.2.3. We have calculated average values of Accuracy, Precision, Sensitivity, Specificity, FPR and FNR in Table 5.2.1 and we can see that DenseNet201 got the highest accuracy of 99.06%.

Class Name	Accuracy	Precision	Sensitivity	Specificity	FPR	FNR
Aphids	97.85%	70.59%	63.16%	99.07%	0.93%	36.84%
Brinjal Shoot and Fruit Borer	99.64%	100%	95.83%	100%	0.0%	4.17%
Cercospora	98.75%	96.97%	92.75%	99.59%	0.41%	7.25%
Flea Beetle	98.39%	84.85%	87.5%	99.05%	0.95%	12.5%
Fruit Rot	99.11%	95%	82.61%	99.81%	0.19%	17.39%
Leaf Roller	100%	100%	100%	100%	0.0%	0.0%
Leaf curleaf	99.82%	96.77%	100%	99.81%	0.19%	0.0%
Leafhopper	99.82%	95.45%	100%	99.81%	0.19%	0.0%
Mealybug	98.75%	92.19%	96.72%	99%	1%	3.28%
Normal Fruit	98.39%	86.67%	98.11%	98.42%	1.58%	1.89%
Normal Leaf	99.46%	97.96%	96%	99.8%	0.2%	4%
Powdery mildew	99.64%	100%	60%	100%	0.0%	40%
Spider Mite	100%	100%	100%	100%	0.0%	0.0%
Spotted Beetle	98.75%	90%	95.74%	99.02%	0.98%	4.26%
Thrips	98.39%	82.35%	70%	99.44%	0.56%	30%
Whitefly	98.21%	58.33%	58.33%	99.09%	0.91%	41.67%

Table 5.2.1: Class Wise Performance Evaluation Matrix of DenseNet201

Class Name	Accuracy	Precision	Sensitivity	Specificity	FPR	FNR
Aphids	97.67%	66.67%	63.16%	98.89%	1.11%	36.84%
Brinjal Shoot and Fruit Borer	99.28%	94%	97.92%	99.41%	0.59%	2.08%
Cercospora	98.93%	95.65%	95.65%	99.39%	0.61%	4.35%
Flea Beetle	98.93%	96.43%	84.38%	99.81%	0.19%	15.62%
Fruit Rot	98.93%	90.48%	82.61%	99.63%	0.37%	17.39%
Leaf Roller	99.64%	100%	75%	100%	0.0%	25%
Leaf curleaf	100%	100%	100%	100%	0.0%	0.0%
Leafhopper	99.46%	90.91%	95.24%	99.63%	0.37%	4.76%
Mealybug	98.03%	87.88%	95.08%	98.39%	1.61%	4.92%
Normal Fruit	99.64%	96.36%	100%	99.6%	0.4%	0.0%
Normal Leaf	99.64%	98%	98%	99.8%	0.2%	2%
Powdery mildew	99.28%	66.67%	40%	99.82%	0.18%	60%
Spider Mite	99.46%	95.31%	100%	99.4%	0.6%	0.0%
Spotted Beetle	98.57%	93.33%	89.36%	99.41%	0.59%	10.64%
Thrips	98.57%	87.5%	70%	99.63%	0.37%	30%
Whitefly	98.57%	62.5%	83.33%	98.9%	1.1%	16.67%

Table 5.2.2: Class Wise Performance Evaluation Matrix of Xception

Class Name	Accuracy	Precision	Sensitivity	Specificity	FPR	FNR
Aphids	98.57%	78.95%	78.95%	99.26%	0.74%	21.05%
Brinjal Shoot and Fruit Borer	99.28%	94%	97.92%	99.41%	0.59%	2.08%
Cercospora	97.67%	84.15%	100%	97.35%	2.65%	0.0%
Flea Beetle	98.75%	93.1%	84.38%	99.62%	0.38%	15.62%
Fruit Rot	98.21%	80.95%	73.91%	99.25%	0.75%	26.09%
Leaf Roller	99.64%	80%	100%	99.64%	0.36%	0.0%
Leaf curleaf	99.82%	100%	96.67%	100%	0.0%	3.33%
Leafhopper	100%	100%	100%	100%	0.0%	0.0%
Mealybug	98.03%	96.3%	85.25%	99.6%	0.4%	14.75%
Normal Fruit	99.11%	94.44%	96.23%	99.41%	0.59%	3.77%
Normal Leaf	99.46%	96.08%	98%	99.61%	0.39%	2%
Powdery mildew	99.28%	100%	20%	100%	0.0%	80%
Spider Mite	99.28%	100%	93.44%	100%	0.0%	6.56%
Spotted Beetle	98.21%	87.76%	91.49%	98.83%	1.17%	8.51%
Thrips	98.57%	77.27%	85%	99.07%	0.93%	15%
Whitefly	98.93%	80%	66.67%	99.63%	0.37%	33.33%

Table 5.2.3: Class Wise Performance Evaluation Matrix of ResNet152V2

Model Name	Accuracy	Precision	Sensitivity	Specificity	FPR	FNR
DenseNet201	99.06%	99.45%	87.30%	99.49%	0.51%	12.70%
Xception	99.04%	88.86%	85.61%	99.48%	0.52%	14.39%
ResNet152V2	98.93%	90.19%	85.49%	99.42%	2.81%	14.51%

Table 5.2.4: Class Wise Performance evaluation metrics using three models

CHAPTER 6
IMPACT ON SOCIETY, ENVIRONMENT, AND ETHICAL ASPECTS

6.1 Impact on Society

Our approach of classifying eggplant disease has a significant role in society. This method can be used as a role model in the automated disease detection field. Also, farmers can be beneficial by using this process which is mostly accurate for fourteen major classifications of disease of eggplant. As eggplant is a common vegetable in our country and plays a vital role in economic development, this process can reduce farmers' environmental barriers of growing more healthy vegetables rather than depending on experts.

6.2 Impact on Environment

Recognizing diseases at an early stage can save a vast amount of plants and fruits, which will also be a great initiative to protect our environment. Saving plants in early-stage has an advantage. Besides fruits or crops, they can produce oxygen to fight the greenhouse effect. So, we think our approach will help the environment by developing the rate of growth in plants.

6.3 Ethical Aspects

Our classification techniques for eggplant disease detection can be implemented without any side effects rather than being friendly for our environment. Disease recognition might result in higher agricultural production with less waste, which could reduce the need for pesticides in some areas.

CHAPTER 7
CONCLUSION, LIMITATIONS AND FUTURE SCOPE

7.1 Conclusion

Our research is based on Eggplant Disease Classification, where we introduced 14 kinds of diseases of eggplant. This is the first attempt to include so many classified diseases of eggplant in Bangladesh so far. Our main target is to accurately predict what disease is affecting an eggplant based on Asian subcontinent weather. That’s why we have included as many diseases as possible. As a result, we have worked with the eggplant diseases named Aphids, Eggplant fruit and shoot borer, Cercospora, Flea Beetle, Fruit rot, Leaf curl, Leaf roller, Leafhopper, Mealybug, Powdery mildew, Spider mite, Spotted beetle, Thrips and Whitefly. After collecting the raw images, we implement augmentation techniques called Image resizing, Rescaling, Rotation, Shearing, Zooming, Flipping, and Fill Mode**.** After that, we use CNN based Transfer Learning models to fit our data into various models called DenseNet201, Xception, ResNetV2. For our proposed Eggplant Disease Classification, we obtained 99.06% accuracy for DenseNet201, 99.04% for Xception, and 98.93% for ResNet152V2. We found that the DenseNet201 transfer learning model achieved the highest accuracy with 99.06% among those three approaches.

7.2 Limitations

Through working with this approach, we experienced some limitations in the field of collecting raw images. As we desired to make as many classifications as possible for all valid eggplant diseases, we needed a lot of raw images of all classified diseases. But we couldn’t collect that much data from the field as some diseases are rare in Bangladesh. For this, we have used some images from the internet, and somehow it damages our particular accuracy.

7.3 Future Scope

For the upcoming days, we decided to develop our research's attributes. We will make it more efficient by adding new datasets and expanding its classification by adding more awaited diseases. We will try to fit this into a learning feature that will enhance its learning capability. We will try this approach with some underrated vegetables of our country to enrich our working area with the Convolutional Neural Network (CNN).

REFERENCES

[1] “YearBook of Agricultural Statistics of Bangladesh 2020”, Bbs.gov.bd, 2021, available at <<http://www.bbs.gov.bd/site/page/3e838eb6-30a2-4709-be85-40484b0c16c6/->>, last accessed on 09-06-2021 at 12:15 PM.

[2] M. M. M. Uddin, “Causal relationship between agriculture, industry and services sector for gdp growth in Bangladesh: An econometric investigation,” Journal of Economics and Sustainable Development, vol. 8, pp. 124-129, 2015.

[3] Bangladesh | FAO Regional Office for Asia and the Pacific | Food and Agriculture Organization of the United Nations", Fao.org, 2021, available at <<http://www.fao.org/asiapacific/perspectives/agricultural-statistics/global-strategy/results-in-the-region/bangladesh/en/>>, last accessed on 09-06-2021 at 12:00 PM.

[4] Rezvi, Minhazur Rahman. “The Factors of Declining Agricultural Growth In Bangladesh And Its Impact On Food Security,” South Asian Journal Of Social Studies And Economics, pp. 1-9, 2018.

[5] Bt Eggplant Increases Farmers' Yields, Cuts Production Costs, Science Speaks, 2021, available at <<https://www.isaaa.org/blog/entry/default.asp?BlogDate=6/2/2021>>, last accessed on 08-06-2021 at 06:12 PM.

[6] PEST RISK ANALYSIS (PRA) OF BRINJAL IN BANGLADESH, available at <<http://plantdiseaseclinic.com/agri-files/PRA-of-Brinjal.pdf>>, last accessed on 15-08-2021 at 05:25 PM.

[7] Double Diamond Farms, available at <<https://www.doublediamondacres.com/2018/02/15/7-surprising-health-benefits-of-eating-eggplant/>>, last accessed on 11-30-2021 at 06:12 PM.

[8] WebMD: Better Information. Better Health, available at <<https://www.webmd.com/food-recipes/eggplant-health-benefits>>, last accessed on 11-30-2021 at 7:19 PM.

[9] Aravind Krishnaswamy R., Purushothaman Raja, Rajendran Ashiwin, and Mukesh Konnaiyar, “Disease Classification in Solanum Melongena Using Deep Learning,” Spanish Journal of Agricultural Research, vol. 17, no. 3, pp. E0204, November 2019.

[10] Krishnaswamy Rangarajan Aravind. Purushothaman Raja, “Disease Classification in Eggplant Using Pre-trained VGG16 and MSVM,” Scientific Reports, vol. 10, no. 1, pp. 2322, January 2020.

[11] Xie, Chuanqi, and Yong He, "Spectrum and Image Texture Features Analysis for Early Blight Disease Detection on Eggplant Leaves," Sensors, vol. 16, no. 5, pp. 676, May 2016.

[12] Sabrol, Hiteshwari, and Satish Kumar, "Plant Leaf Disease Detection Using Adaptive Neuro-Fuzzy Classification," Advances in Intelligent Systems and Computing, vol. 943, pp. 434–443, January 2020.

[13] R Anand, S Veni, and J Aravinth, "An Application of Image Processing Techniques for Detection of Diseases on Brinjal Leaves Using K-Means Clustering Method," International Conference on Recent Trends in Information Technology (ICRTIT), pp. 1-6, September 2016.

[14] Jake M., "Mobile-Based Eggplant Diseases Recognition System Using Image Processing Techniques," International Journal of Advanced Trends in Computer Science and Engineering, vol. 9, no. 1.1, pp. 182–190, February 2020.

[15] D. Wu, L. Feng, C. Zhang, Y. He, "Early Detection of Botrytis Cinerea on Eggplant Leaves Based on Visible and near-Infrared Spectroscopy," Transactions of the ASABE, vol. 51, no. 3, 2008, pp. 1133–1139, 2008.

[16] M. T. Habib, M. J. Mia, Uddin, M. S., and F. Ahmed, "An in-depth exploration of automated jackfruit disease recognition," Journal of King Saud University - Computer and Information Sciences, May 2020.

[17] M. J. MIA, S. K. MARIA, S. S. TAKI, and A. A. BISWAS, "Cucumber disease recognition using machine learning and transfer learning," Bulletin of Electrical Engineering and Informatics, vol.10, no. 6, pp. 3432-3443, December 2021

[18] P. Sharma,Y. P. S. Berwal, and W. Ghai, "Performance Analysis of Deep Learning CNN Models for Disease Detection in Plants Using Image Segmentation," Information Processing in Agriculture, vol. 7, no. 4, pp. 566–574, November 2019.

[19] P. Revathi, and M. Hemalatha, "Identification of cotton diseases based on cross information gain_deep forward neural network classifier with PSO feature selection," International Journal of Engineering and Technology (IJET), vol. 5 no. 6, pp. 4637-4642, 2013.

[20] M. Jafari, S. Minaei, and N. Safaie, "Detection of Pre-Symptomatic Rose Powdery-Mildew and Gray-Mold Diseases Based on Thermal Vision," Infrared Physics & Technology, vol. 85, pp. 170–183, May 2017.

[21] R. Zhou, S. Kaneko, F. Tanaka, M. Kayamori, and M. Shimizu, “Disease Detection of Cercospora Leaf Spot in Sugar Beet by Robust Template Matching,” Computers and Electronics in Agriculture, vol. 108, 2014, pp. 58–70, October 2014.

[22] Roboflow Blog, available at <<https://blog.roboflow.com/why-preprocess-augment/>>, last accessed on 11-30-2021 at 08:32 PM.

[23] Binary Image classifier CNN using TensorFlow, available at <<https://medium.com/techiepedia/binary-image-classifier-cnn-using-tensorflow-a3f5d6746697>>, last accessed on 11-30-2021 at 09:10 PM.

[24] Understanding and visualizing DenseNets, available at <<https://towardsdatascience.com/understanding-and-visualizing-densenets-7f688092391a>>, last accessed on 11-30-2021 at 09:21 PM.

[25] DenseNet, available at <<https://keras.io/api/applications/densenet/>>, last accessed on 11-30-2021 at 09:24 PM.

[26] Densely Connected Convolutional Networks, available at <<https://arxiv.org/abs/1608.06993>>, last accessed on 11-30-2021 at 09:41 PM.

[27] Review: Xception — With Depthwise Separable Convolution, Better Than Inception-v3 (Image Classification), available at <<https://towardsdatascience.com/review-xception-with-depthwise-separable-convolution-better-than-inception-v3-image-dc967dd42568>>, last accessed on 11-30-2021 at 09:45 PM.

[28] Xception, available at <<https://keras.io/api/applications/xception/>>, last accessed on 11-30-2021 at 09:51 PM.

[29] ResNet and ResNetV2, available at <<https://keras.io/api/applications/resnet/#resnet152v2-function>>, last accessed on 11-30-2021 at 10:11 PM.

[30] Limitless Data Science, available at <<https://limitlessdatascience.wordpress.com/2019/05/03/residual-neural-network-resnet/>>, last accessed on 11-30-2021 at 10:18 PM.

[31] Understanding Confusion Matrix, available at <<https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62/>>, last accessed on 11-30-2021 at 10:55 PM.

[32] M. Rahman, S. A. Prodhan, M. J. Mia, M. T. Habib, and F. Ahmed, “Pigeon Breed Recognition using Convolutional Neural Network,” Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV 2021), pp. 1426-1430, May 2021

[33] J. Mia, H. Bijoy, Uddin S. Uddin, D. Raza, “Real-Time Herb Leaves Localization and Classification Using YOLO,” 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-7, November 2021

[34] S. K. Maria, S. S. Taki., M. J. Mia, A. A. Biswas., A. Majumder, F. Hasan, “Cauliflower Disease Recognition Using Machine Learning and Transfer Learning,” Innovations in Computing. Smart Innovation, Systems and Technologies, vol. 235, pp. 359-375, 2021

An Approach For Eggplant Disease Recognition

ORIGINALITY REPORT

25%	16%	16%	12%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	3%
2	Submitted to Daffodil International University Student Paper	3%
3	Md. Jueal Mia, Syeda Khadizatul Maria, Shahrun Siddique Taki, Al Amin Biswas. "Cucumber disease recognition using machine learning and transfer learning", Bulletin of Electrical Engineering and Informatics, 2021 Publication	2%
4	Submitted to University of Essex Student Paper	1%
5	www.ncbi.nlm.nih.gov Internet Source	1%
6	"Smart Systems: Innovations in Computing", Springer Science and Business Media LLC, 2022 Publication	1%
7	Kanthan Muthukannan, Pitchai Latha. "A GA_FFNN algorithm applied for classification	1%