Identification of Cucumber Leaf Disease via Deep Learning Based Classification

Mizanur Rahman[1], Q. Mohona[2]
Department Of CSE, Daffodil International University
[2]R Institute of Information Technology (IIT), University of Dhaka, Dhaka, Bangladesh
[2]R Institute of Information Technology (IIT), University of Dhaka, Dhaka, Bangladesh

Article Info

Article history:

Received month dd, yyyy
Revised month dd, yyyy
Accepted month dd, yyyy

Keywords:

Deep Learning
Data Augmentation
Third keyword
Fourth keyword
Fifth keyword

ABSTRACT

Cucumber is a cash crop in Bangladesh as it is a side dish grown commercially in cultivable lands year-round. The early prediction of disease-prone crops could save grooming time and minimize losses. The conventional method of examining leaves just through observation of the human eye could only detect the diseases at an advanced stage without a concrete decision of which disease it might be and regular inspection is labour intensive, inaccurate and often unreliable. This study aims to evaluate available machine learning-based image analyses to distinguish between healthy and ailed leaves of cucumbers by observable traits detected and retrieved by training the machine learning models as well as identifying the disease through classification.

To fulfil this purpose, CNN, InceptionV3, and EfficientNetB4 are the models implemented in this paper to improve the classification of objects. Furthermore, a comparative analysis was done regarding the methods to determine the technique that yields the best performance. The dataset was optimized by pre-processing techniques and irrelevant data was removed as part of the data-cleaning process, and the leaves are classified into four categories, namely Angular leaf spot, Downy mildew, Powdery mildew, and Good leaf. The EfficienNetB4 model achieved the highest accuracy reaching an accuracy rate of 86%. However, a comparative analysis of the available models was conducted in this paper to reach a solid decision. This research investigates the data collection on cucumber diseases addressing issues like seasonal variation and categorization of the images via preprocessing and retrieving accurate parameters to determine categories and evaluation of machine learning models upon the processed dataset.

Corresponding Author:

Quazi Mahfuza Tasnim Mohona
Institute of Information Technology (IIT), University of Dhaka, Dhaka, Bangladesh
Suhrawardi Udyan Rd, Dhaka 1200
Email: qmohona@mail.com
Email: mizanurrahman.cse@diu.edu.bd

1. INTRODUCTION (10 PT)

A crucial indication of early success in agriculture depends upon the farmer's ability to early detection of diseases which could substantially assist in the prevention of further progression of the disease. Hence, it is imperative that not only there are measures in place for the early detection of such ailments in crops but also detection of the particular disease at play so that the parties concerned maybe able to take steps to cure the particular ailment which would determine the wider success of the crop.

Investigation of machine learning-based image analysis for accurate diagnosis of cucumber disease via determination of the best technique through comparative analysis was the focus of this research.

Implementation of several machine learning models in disease detection was one of the highlights of this research, as it would eventually contribute to accurate diagnosis of the crop in question to alleviate further deterioration and widespread prevention.

Raising agricultural output demands prompt and accurate diagnosis of diseases followed by remedies and the only way to ensure correct measures is to precisely detect the ailment of the crop in question. Cucumis sativus, that is commonly known as the cucumber, is a popular vegetable that has several health advantages. Several leaf diseases significantly affect crop quality and production of this popular food often critically damaging the crop during cultivation.

2. LITERATURE REVIEW

Early diagnosis and treatment of these diseases are critical for minimizing losses and implementing timely management strategies. Cucumber leaf diseases are classified into seven categories and training images in machine learning methods using K-means clustering and Random Forest provides an accuracy of 89.93% [1]. Additionally, the study addresses the performance of InceptionV3, MobileNetV2, and VGG16. The MobileNetV2 achieved the highest success rate of 93.23%. The study stresses the importance of finding diseases early on to achieve the highest possible harvest.

Computer-aided approaches for early disease classification in salad cucumber leaves, with an emphasis on smart organic farming difficulties in India [2] were addressed by combining Support Vector Machine (SVM) and K-Means Clustering enabling the existing trend of machine learning-based image processing for precision agriculture. The system possesses practical viability for widespread agricultural benefit as it has demonstrated efficacy in identifying diseases in crops such as paddy leaves and cotton.

[3]tackles important issues in real-world plant disease identification with IoT. The suggested approach, which combines two-stage segmentation, rotation, translation, AR-GAN-based augmentation, and DICNN, outperforms existing techniques with remarkable average identification accuracies of 96.11% and 90.67% on lesion and raw field diseased leaf datasets. This is achieved using deep convolutional neural networks (CNNs) and a small sample size. A major step towards the actual application of advanced technology in agricultural IoT systems and precision agriculture, the work is in line with the emerging trend of using CNNs and GANs for strong plant disease recognition.

The issue in the detection of automated disease classification in cucumbers needs to be handled in a time-efficient manner—which is essential to maintaining agricultural productivity—it is tackled in this work. The suggested architecture combines VGG19, Inception V3, deep learning, and a unique parallel maximum correlation fusion technique to reach an impressive 96.5% accuracy. When compared to more current methods, it emphasizes the value of sophisticated computational methods for illness diagnosis. I investigate existing techniques such as feature selection, dilated convolution, SVM, and sparse representation. With its optimization of deep features by Whale Optimization, reduction of computing time, and improvement of disease recognition efficiency, the study makes a significant contribution to the field of deep learning-based precision agriculture. [4]

The study in [5] tackles the critical problem of identifying cucumber diseases, which is essential to maintaining agriculture. With an Entropy-ELM-based architecture and deep learning, the study attains an impressive 98.48% accuracy. Comparisons with current methods highlight the superiority of the suggested method. Previous studies that use techniques such as feature selection (Entropy ELM), pre-trained models (VGG19, Inception V3), and data augmentation are presented. There are noteworthy initiatives that use a variety of models (DeepLabV3+, U-Net, EfficientNet, methodologies (WOA-based feature selection), and methods (SHSB saliency, probability distribution-based entropy). By combining these methods, this article contributes and highlights the value of sophisticated computational methods for disease classification and classification in precision agriculture [5].

The body of [6] research highlights a shift towards reliable, automated techniques, such as GPDCNN, which hold the promise of better crop disease classification, especially with irregularly diseased leaf images' complexity.

The study employs a unique dataset [7]

Convolutional Neural Network (CNN) achieves a remarkable accuracy (98.29% for training and 98.029% for testing) with a focus on tomatoes, peppers, and potatoes of Plant Village for testing) While [8] focuses on Tomatoes, peppers and potatoes. The study [9] uses the Faster R-CNN model, a Convolutional Neural Network-based object classification technique and acquired a respectable 94.86% accurate classification rate. [10] A hybrid framework achieves 93.50% accuracy in cucumber disease classification, emphasizing real-time applications. Sparse representation and clustering achieve 85.7% recognition, addressing automated crop disease classification challenges [11]. [12] CNN-based segmentation achieves 96.08% accuracy in powdery mildew assessment, crucial for breeders. [13] The fusion-based method achieves 94.30% accuracy in disease identification, outperforming others. [14] DCNN achieves 93.4% accuracy in cucumber disorder classification, superior to traditional methods. [15] Image processing and ANN achieve 80.45% accuracy in early crop illness diagnosis. [16] ML and image processing highlight the importance of early disease identification in cucumbers. [17] Autonomous device using ANN achieves accurate detection, suggesting further research for real-time capabilities. [18] The machine learning method achieves a 96.80% success rate in cucumber disease classification. [19] Deep learning system achieves 65.8% precision in wide-angle plant disease diagnostics.

CNN classifier achieves 95.5% accuracy in cucumber disease diagnosis, advancing practical plant diagnosis [20]. Two-stage model achieves 93.27% segmentation accuracy for cucumber leaf disease severity assessment. [21]. [22] The SVM-based technique improves cucumber disease recognition, offering insights for efficient diagnosis. The investigation of [23] Support Vector Machine (SVM) suggests a novel experimental strategy that uses individual leaf spots as data. The optimal performance is achieved with the suggested sampling strategy and the RBF kernel function, as demonstrated by comparative testing using various kernel functions. The work improves the accuracy of classifying cucumber diseases and makes automated illness identification possible. The research [24] addresses the insufficient training samples and imperfect measurement settings by implementing the Extended Collaborative Representation (ECR) model for cucumber leaf disease classification with a 94.7% diagnostic accuracy. The [25] overcomes the difficulties of manual diagnosis by a tailored Convolutional Neural Network (CNN) for the recognition of diseases after the creation of a new dataset. The authors show that it outperforms pre-trained models with a high recognition accuracy of 98.19%.

3. RESEARCH METHODOLOGY

The Deep Learning-Based Cucumber Leaf Disease Classification for Precision Agriculture commences with the collection of datasets, preprocessing of the collected datasets and finally implementation of machine learning mechanisms.

3.1 DATA ACQUISITION PROCEDURE

A varied set of high-definition photos of cucumber leaves that covered a wider range of growth phases, in infected and healthy conditions along with varied lighting circumstances were taken into consideration as a dataset. We made sure that the dataset was representative of common cucumber leaf diseases that affect cucumbers. We accurately labelled the dataset with the presence of the particular diseases in consideration for them to be passed onto the supervised training phase.

The collection of datasets on Cucumber diseases was another significant contribution of this research. The data collection for this research was done in two seasons to cover all the bases since Cucumbers grow and are affected by distinct weather characteristics The photos were collected via the Mobile Phone Samsung A50 which possesses the specifications of a 48 MP primary camera, f/2.0, 26mm (wide), 1/2.0", 0.8µm, PDAF, 8 MP, f/2.2, 13mm (ultrawide), 1/4.0", 1.12µm, 5 MP, f/2.2, (depth), LED flash, panorama, HDR, 4K@30fps 1080p@30fps., Various cucumber fields farmers were visited for the collection of data. A total of 517 images were acquired out of which there are 98 images of Angular Leaf Spot, 106 images of Downy Mildew and 236 images of Good Leaf. The ratio of the dataset is demonstrated in Figure 1.

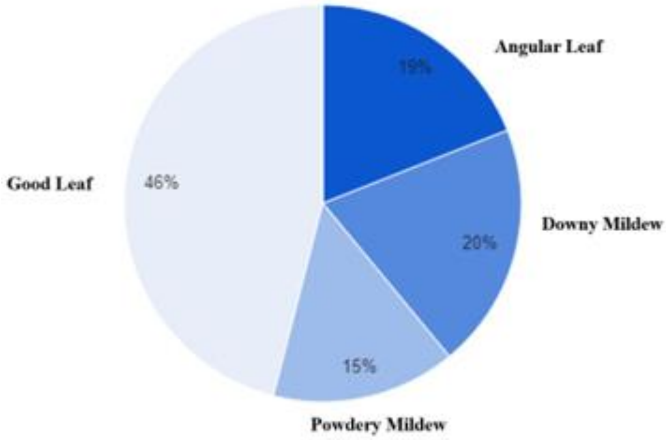


Figure 1. Ratio of Dataset

3.2 DATA PREPROCESSING

The dataset contains four classes of images namely: Angular leaf Spot, Downy Mildew, Powdery Mildew, and good leaf. All images were shaped in 256×256 dimensions and then the image dataset was augmented into training, testing and validation respectively by 70%, 15% and 15%. Figure 2 demonstrates the augmented images in 2D arrangement to add more images to the acquired dataset.

The data were normalized and scaled for further augmentation as a preparation for training the model. To avoid overfitting, set up the architecture with the proper layers, activation functions, and regularization strategies. Table 1. shows the description of the dataset after augmentation. There are 1473 images in the dataset which are split into four classes.

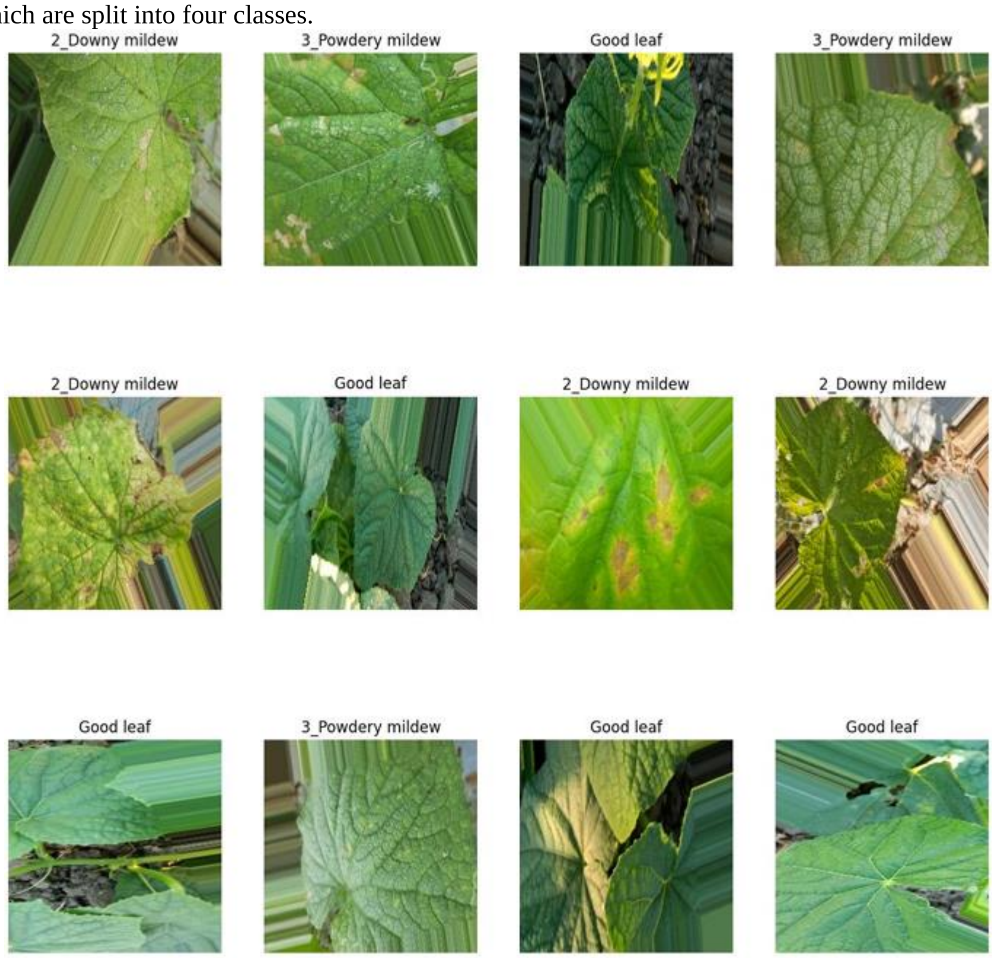


Figure 2. Preprocessed Images

Table 1. Description of datasets

Dataset	Classes	Original Images	Augmented Images
Real Dataset	Angular Leaf Spot	98	192
	Downy Mildew	106	173
	Powdery Mildew	77	151
	Good leaf	236	440
	Total	517	956

3.3 PROPOSED METHODOLOGY

The experiment was a multifaceted initiative and the stages of the experiment are discussed in the following section. For this experiment, Customized Convolutional Neural Network (CNN) 1 and 2, InceptionV3 and EfficientNetB4 models were used. A count of 25 Epochs was initiated for CXNN 1 and 2, 50 epochs for Inception V3 and 75 epochs for EfficientNetB4. A sequential architecture along with an accuracy graph is shown in Figure 3, In the figure the summary of the model CNN-1 is demonstrated as 4 convolutional layers are implemented with 4 pooling layers and a fully connected layer with 512 neurons. The total parameter of this model is 26,081,092. The accuracy of this model is 77.34%.

Convolutional Neural Network is the most popular model for image classification, recognition, and object classification. It's made of three parts. That includes the fully linked layer, the pooling layer, and the convolutional layer

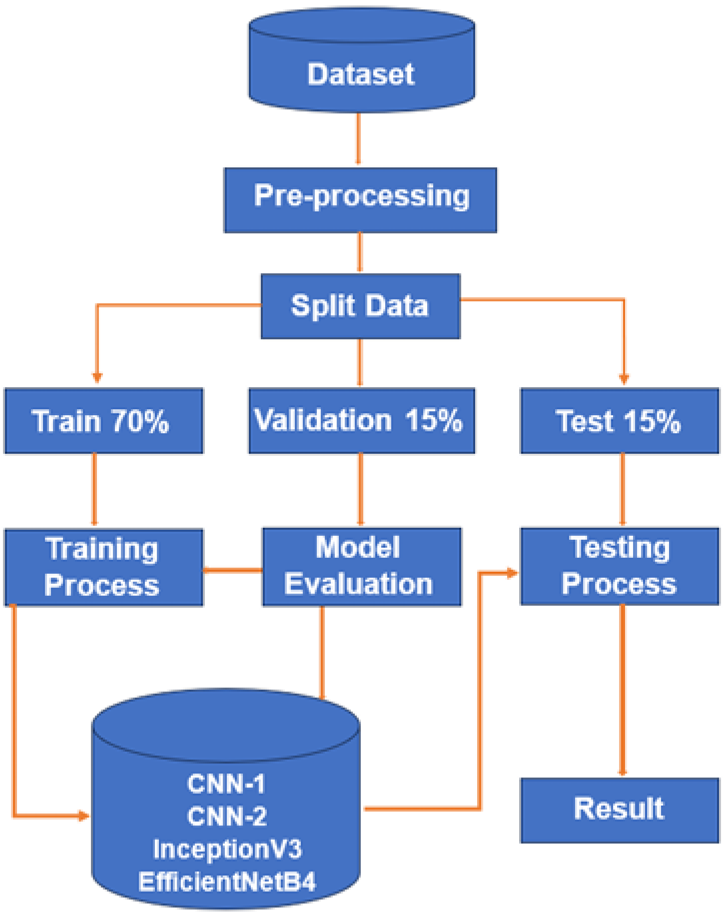


Figure 3. Processed Model

Figure 3 depicts the processed model implemented in this study. Augmenting images is a method of altering data that was used in this paper. The phases of the approach are data collection, preprocessing, data augmentation, model training (CNN-1, CNN-2 InceptionV3, EfficientNetB4), and model testing.

3.3.1 Convolutional Neural Network(CNN)

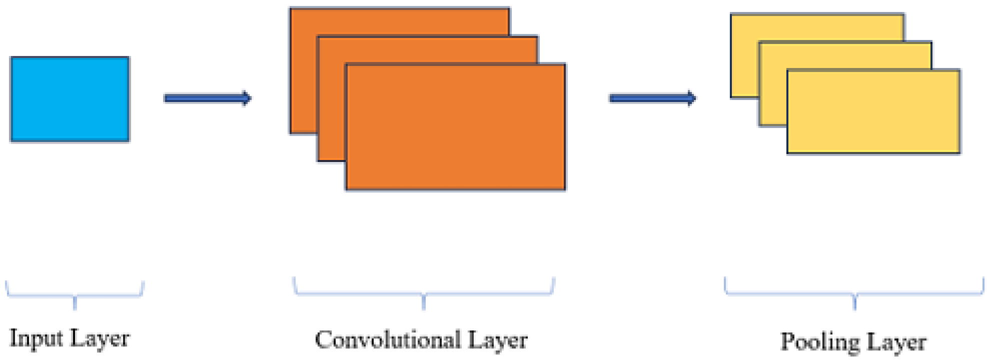


Figure 4: Pooling Layer

The Convolutional layer in CNN divides the images into many filters and extracts some important features like shape, colour, and size. There is also a RELU activation part, which works for doing nonlinear and training time fast. In Figure 4, the pooling layer, after that fully joined layer starts. The last layer is a fully joined layer. It is also called the thick layer. It takes all features from a flattened layer and links all the features from one another. And finally, classify what the picture is and give probabilities of what the image could be.

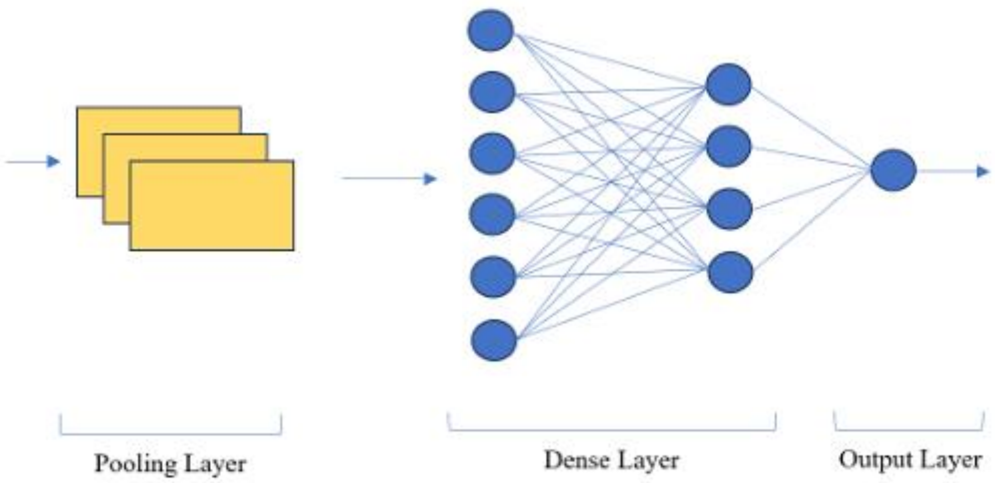


Figure 5: Fully Connected Layer or Dense Layer

In Figure 5, the fully connected layer is shown where CNN classifies or recognizes a picture. the layer number was increased with each change of the RGB picture pixel (example: 32×32) and also changed the neuron number in the thick layer. For the first customized CNN, four convolutional layers are 3×3 kernel and 32×32, 64×64, 128×128, and 256×256pixel values with 512 neurons. For the second customized CNN, four convolutional layers are 3×3 kernel and 32×32, 64×64, 128×128, and 256×256pixel values with 512 neurons. 77.34% accuracy is achieved by the first customized CNN for the Real Capture Image Dataset.

When the number of neurons and pixel value come together perfectly, accuracy increases. When overfitting occurs in the CNN model, accuracy decreases. The second customized CNN model increased the accuracy to 83.59%.

3.3.2 InceptionV3

Figure 6 depicts the architecture of the InceptionV3 Model. InceptionV3. InceptionV3 is the name of a 48-layer pre-trained model which was implemented here for picture categorization. This network repeats blocks, with each block's output serving as the subsequent block's input. Every block is identified as an origin block. InceptionV3 uses RGB images by default and has a 299 × 299 picture size. Block by block, the features of an image are extracted. 71.48% is the accuracy for InceptionV3

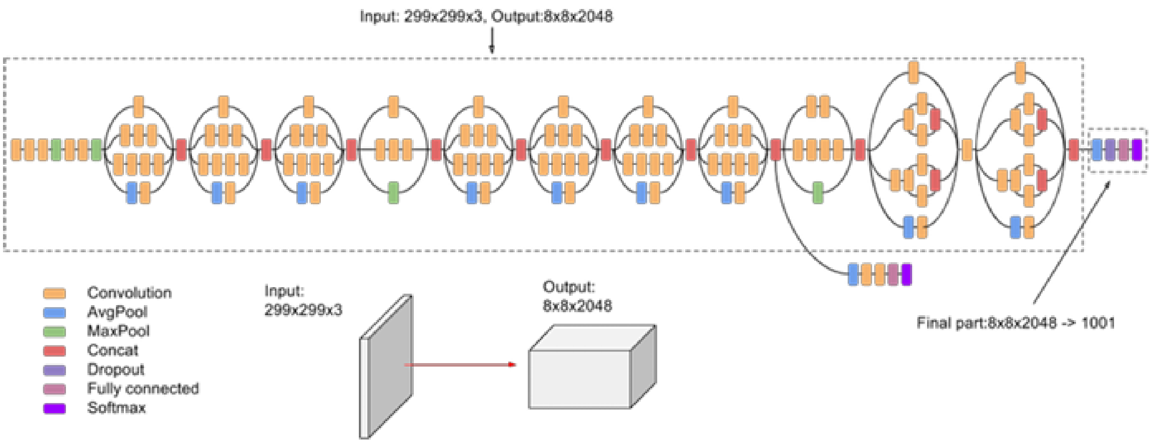


Figure 6: InceptionV3 Architecture.

3.3.3 EfficientNetB4

The base model (B0) is scaled up with a scaling factor (ϕ) of about 4.0 to create EfficientNetB4. Multiplying the number of channels in each layer increases the network's diameter. To effectively capture more intricate patterns in the data, this is an essential component. To obtain more detailed information, the input photographs' resolution is raised. This facilitates the network's ability to identify things at various scales. Figure (3.8) is the architecture of EfficientNetB4. The accuracy of EfficientNetB4 is 86.32%.

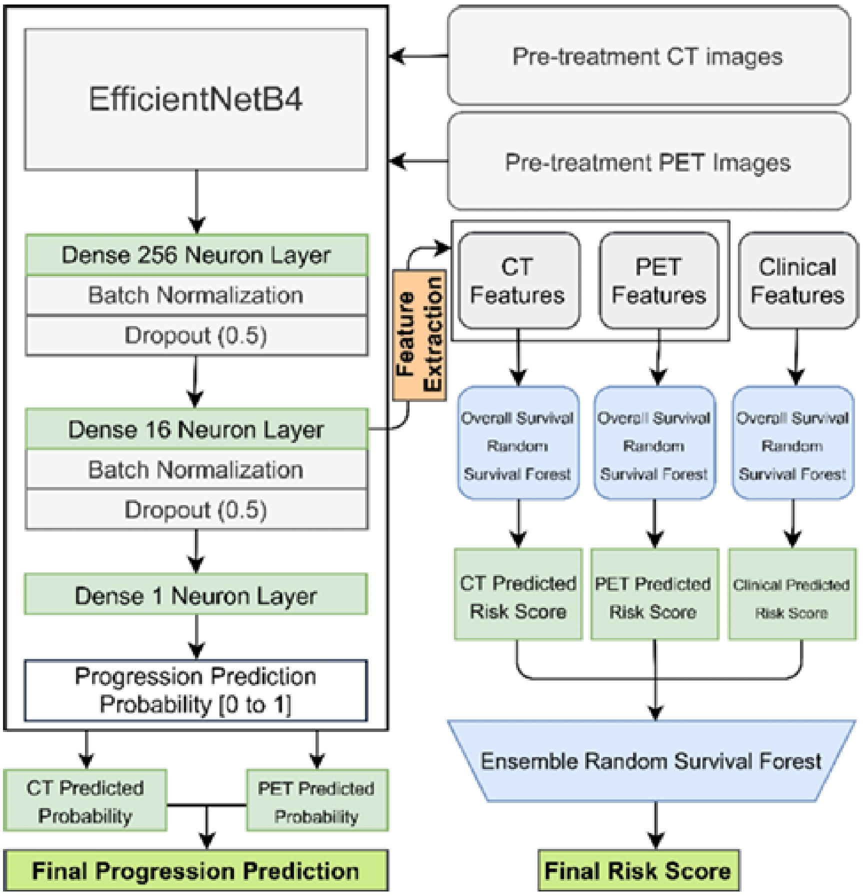


Figure 7: EfficientNetB4 Architecture.

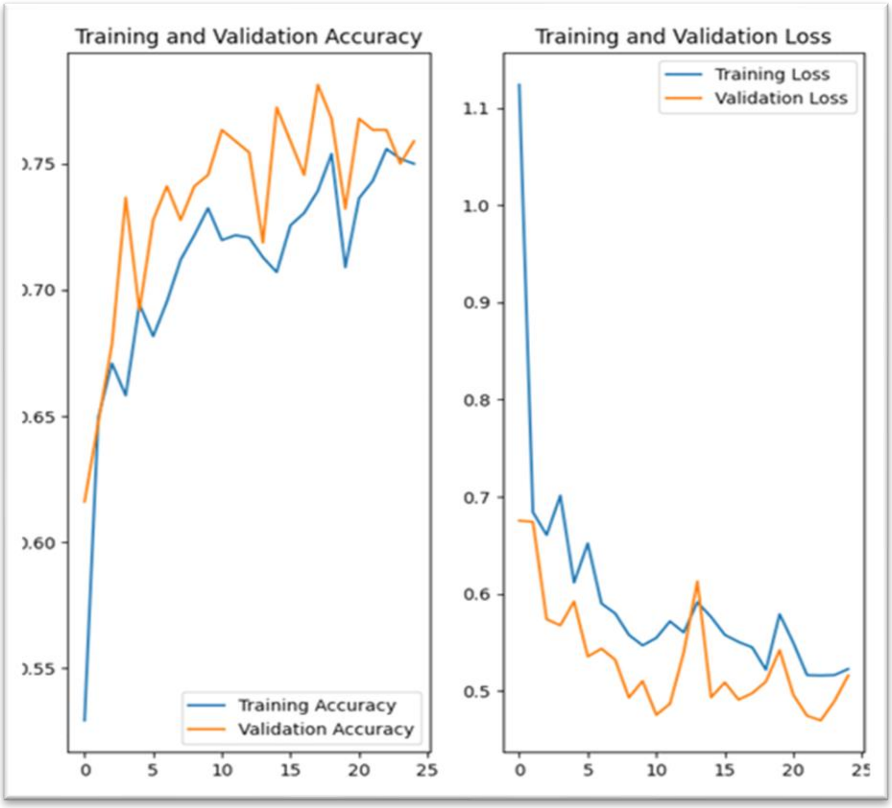
4. RESULTS AND DISCUSSION

Comprehensive research on cucumber leaf disease classification is presented to recognize the plant life. In this research, a customized Convolutional Neural Network (CNN) 1 & 2, InceptionV3, and EfficientNetB4 models are used. Here 25 epochs for CNN 1 & 2, 50 epochs for InceptionV3, and 75 epochs for EfficientNetB4 are deployed. Below are the accuracy graph and the sequential architecture.
For CNN-1:

Layer (type)	Output Shape	Param #
sequential (Sequential)	(32, 256, 256, 3)	0
conv2d (Conv2D)	(32, 254, 254, 32)	896
max_pooling2d (MaxPooling2D)	(32, 127, 127, 32)	0
conv2d_1 (Conv2D)	(32, 125, 125, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(32, 62, 62, 64)	0
conv2d_2 (Conv2D)	(32, 60, 60, 128)	73856
max_pooling2d_2 (MaxPooling2D)	(32, 30, 30, 128)	0
conv2d_3 (Conv2D)	(32, 28, 28, 256)	295168
max_pooling2d_3 (MaxPooling2D)	(32, 14, 14, 256)	0
flatten (Flatten)	(32, 50176)	0
dense (Dense)	(32, 512)	25690624
dense_1 (Dense)	(32, 4)	2052

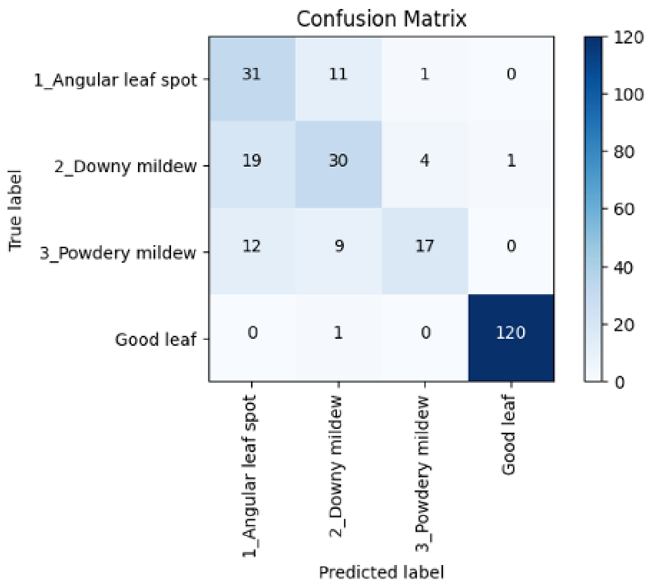
Total params: 26081092 (99.49 MB)
Trainable params: 26081092 (99.49 MB)
Non-trainable params: 0 (0.00 Byte)

(a)



(b)
Figure 8. Model Summary of CNN-1 (a)
(b) Accuracy and Loss Graph of CNN-1

Figure 8 shows the model summary of CNN-1. Here 4 convolutional layers are used, 4 pooling layers and a fully connected layer with 512 neurons. The total parameter of this model is 26,081,092. And the accuracy of this model is 77.34%. The model's accuracy and the loss graph are given below. Figure 8 (b) is the accuracy and loss graph of CNN-1. The training and validation accuracy and loss graphs in the context of the Customized Convolutional Neural Network (CNN-1) offer important insights into how well the model is learning from the data throughout the training process and how it generalizes to new, unseen information.



	precision	recall	f1-score	support
0	0.50	0.72	0.59	43
1	0.59	0.56	0.57	54
2	0.77	0.45	0.57	38
3	0.99	0.99	0.99	121
accuracy			0.77	256
macro avg	0.71	0.68	0.68	256
weighted avg	0.79	0.77	0.77	256

Figure. 9 Confusion Matrix & Classification Report of CNN-1

Figure 9 is the confusion matrix & and classification report of CNN-1. After examining the CNN-1 model classification report and confusion matrix, the accuracy was found to be 77.34%, which is excellent. Additionally, this research has a f1-score of around 0.77. For CNN-2, In this figure (4.4) shows the model summary of CNN-2. Here 4 convolutional layers are used, 4 pooling layers and a fully connected layer with 512 neurons. The total parameter of this model is 26,085,060. And the accuracy of this model is 83.59%. The model's accuracy and loss graph are given below.

Layer (type)	Output Shape	Param #
sequential (Sequential)	(None, 256, 256, 3)	0
conv2d_4 (Conv2D)	(32, 254, 254, 32)	896
max_pooling2d_4 (MaxPooling2D)	(32, 127, 127, 32)	0
batch_normalization (BatchNormalization)	(32, 127, 127, 32)	128
conv2d_5 (Conv2D)	(32, 125, 125, 64)	18496
max_pooling2d_5 (MaxPooling2D)	(32, 62, 62, 64)	0
batch_normalization_1 (BatchNormalization)	(32, 62, 62, 64)	256
conv2d_6 (Conv2D)	(32, 60, 60, 128)	73856
max_pooling2d_6 (MaxPooling2D)	(32, 30, 30, 128)	0
batch_normalization_2 (BatchNormalization)	(32, 30, 30, 128)	512
conv2d_7 (Conv2D)	(32, 28, 28, 256)	295168
max_pooling2d_7 (MaxPooling2D)	(32, 14, 14, 256)	0
batch_normalization_3 (BatchNormalization)	(32, 14, 14, 256)	1024
flatten_1 (Flatten)	(32, 50176)	0
dense_2 (Dense)	(32, 512)	25690624
dropout (Dropout)	(32, 512)	0
batch_normalization_4 (BatchNormalization)	(32, 512)	2048
dense_3 (Dense)	(32, 4)	2052

Total params: 26085060 (99.51 MB)
Trainable params: 26083076 (99.50 MB)
Non-trainable params: 1984 (7.75 KB)

Figure 10: Model Summary of CNN-2

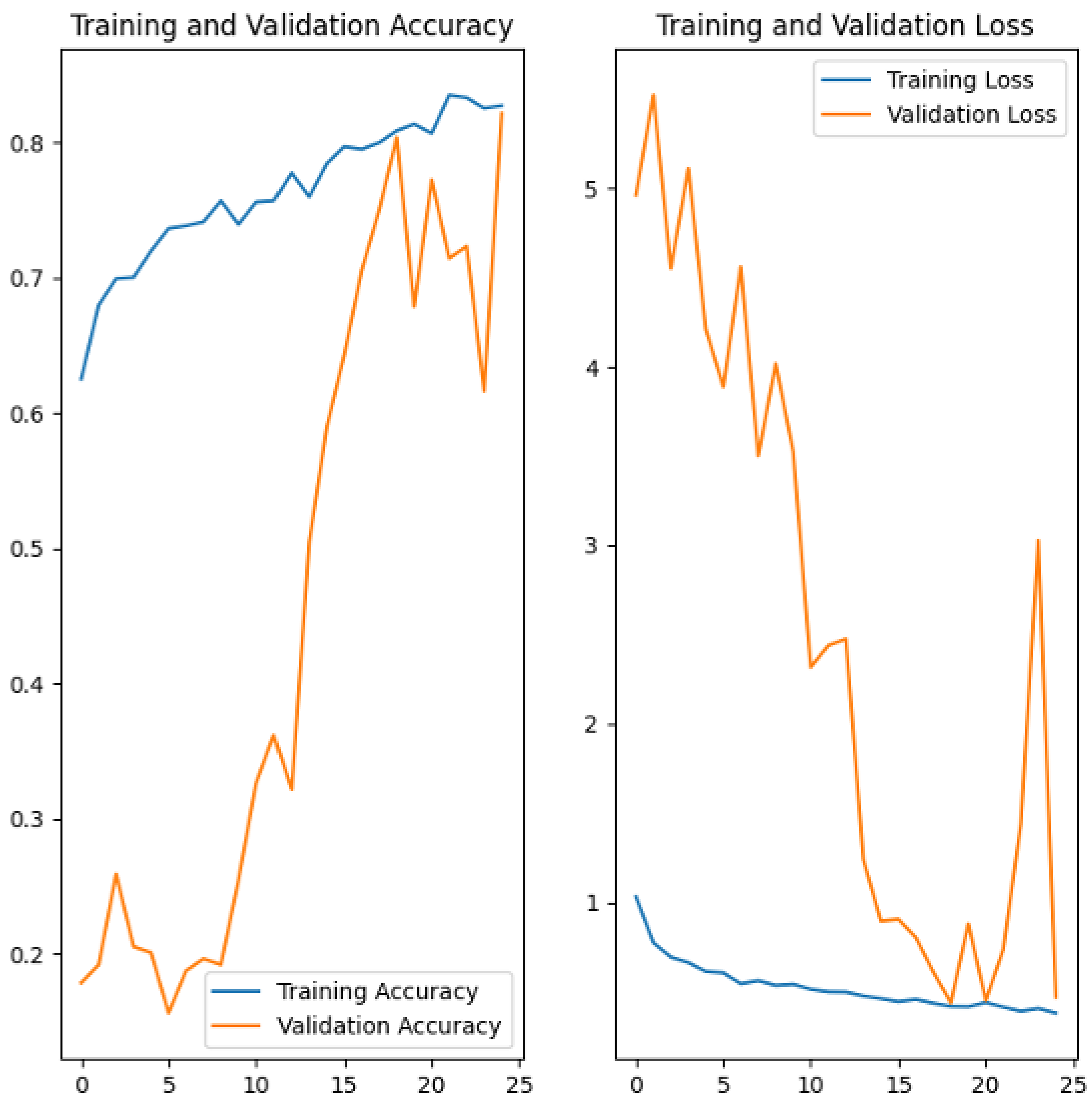
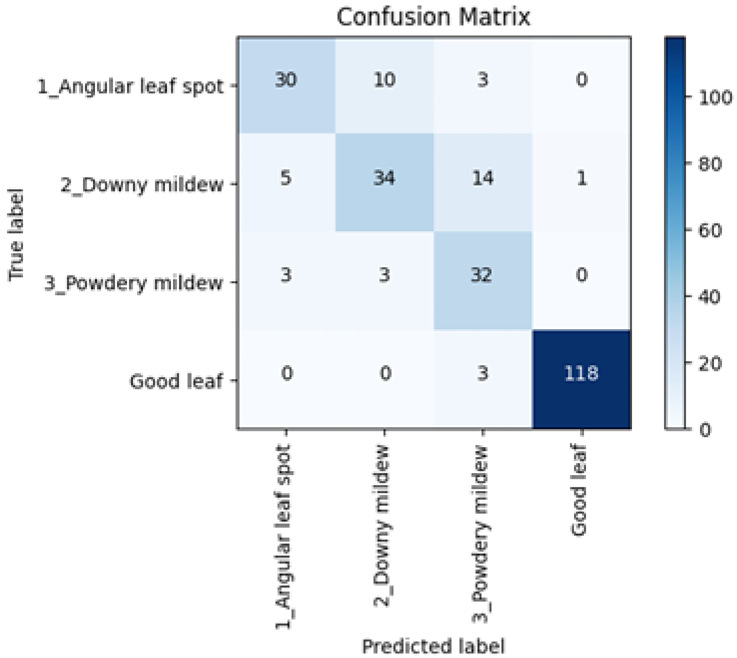


Figure 11.

Figure (4.5) is the accuracy and loss graph of CNN-2. The training and validation accuracy and loss graphs in the context of the Customized Convolutional Neural Network (CNN-2) offer important insights into how well the model is learning from the data throughout the training process and how it generalizes to new, unseen information.



	precision	recall	f1-score	support
0	0.79	0.70	0.74	43
1	0.72	0.63	0.67	54
2	0.62	0.84	0.71	38
3	0.99	0.98	0.98	121
accuracy			0.84	256
macro avg	0.78	0.79	0.78	256
weighted avg	0.85	0.84	0.84	256

Figure 11: Confusion Matrix & Classification Report of CNN-2

Figure 11 is the confusion matrix & and classification report of CNN-2. After examining the CNN-2 model classification report and confusion matrix, the accuracy was found to be 83.59%, which is excellent. Additionally, this research has an f1-score of around 0.84.
For InceptionV3:

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 256, 256, 3)]	0	[]
conv2d_8 (Conv2D)	(None, 127, 127, 32)	864	['input_1[0][0]']
batch_normalization_5 (BatchNormalization)	(None, 127, 127, 32)	96	['conv2d_8[0][0]']
activation (Activation)	(None, 127, 127, 32)	0	['batch_normalization_5[0][0]']
conv2d_9 (Conv2D)	(None, 125, 125, 32)	9216	['activation[0][0]']
batch_normalization_6 (BatchNormalization)	(None, 125, 125, 32)	96	['conv2d_9[0][0]']
activation_1 (Activation)	(None, 125, 125, 32)	0	['batch_normalization_6[0][0]']
conv2d_10 (Conv2D)	(None, 125, 125, 64)	18432	['activation_1[0][0]']
batch_normalization_7 (BatchNormalization)	(None, 125, 125, 64)	192	['conv2d_10[0][0]']
activation_2 (Activation)	(None, 125, 125, 64)	0	['batch_normalization_7[0][0]']
max_pooling2d_8 (MaxPooling2D)	(None, 62, 62, 64)	0	['activation_2[0][0]']
conv2d_11 (Conv2D)	(None, 62, 62, 80)	5120	['max_pooling2d_8[0][0]']
batch_normalization_8 (BatchNormalization)	(None, 62, 62, 80)	240	['conv2d_11[0][0]']
activation_3 (Activation)	(None, 62, 62, 80)	0	['batch_normalization_8[0][0]']
conv2d_12 (Conv2D)	(None, 60, 60, 192)	138240	['activation_3[0][0]']
batch_normalization_9 (BatchNormalization)	(None, 60, 60, 192)	576	['conv2d_12[0][0]']
activation_4 (Activation)	(None, 60, 60, 192)	0	['batch_normalization_9[0][0]']

Figure 12: Model Summary of InceptionV3

This figure (12) shows the model summary of InceptionV3. Here 94 convolutional layers are used, 94 activation layers. The total parameter of this model is 23,116,580. And the accuracy of this model is 71.48%. The model's accuracy and loss graph is given below.

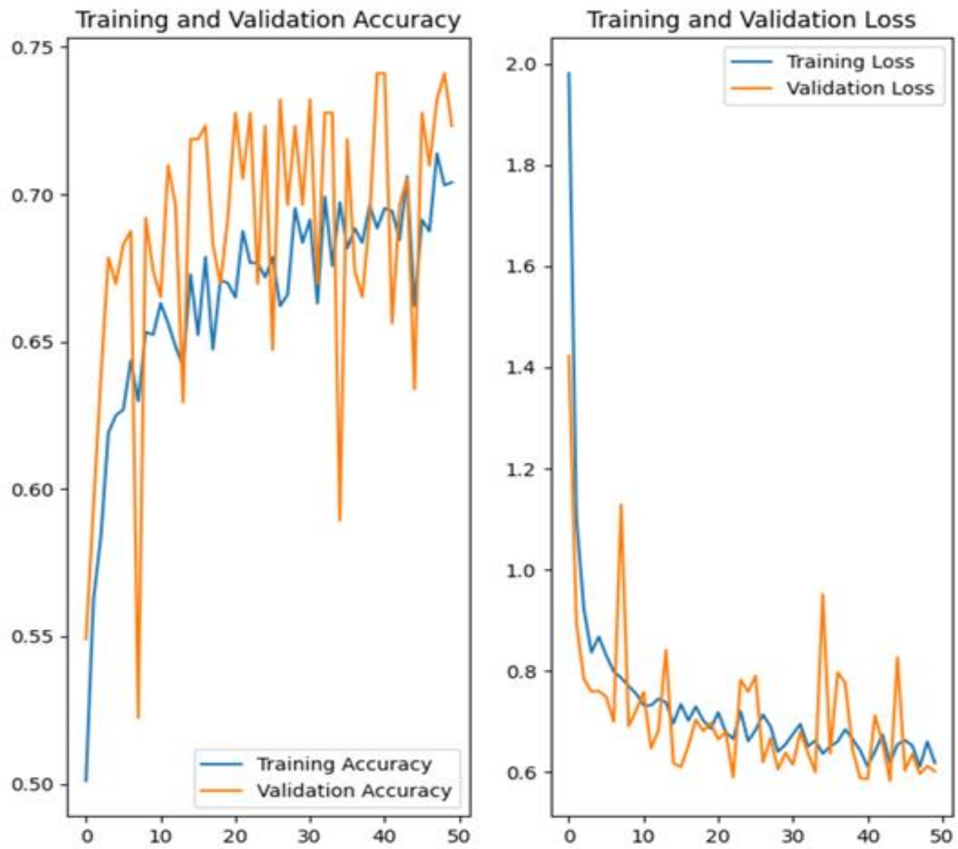
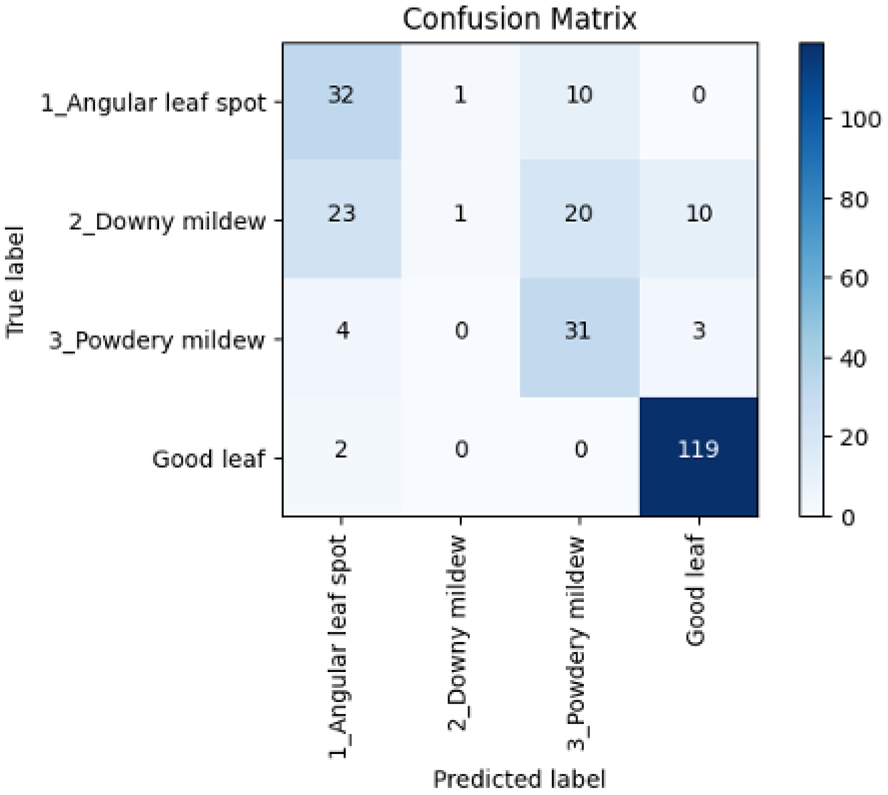


Figure 13: Accuracy and Loss Graph of InceptionV3.

Figure (13) is the accuracy and loss graph of InceptionV3. The training and validation accuracy and loss graphs in the context of the InceptionV3 offer important insights into how well the model is learning from the data throughout the training process and how it generalizes to new, unseen information.



	precision	recall	f1-score	support
0	0.52	0.74	0.62	43
1	0.50	0.02	0.04	54
2	0.51	0.82	0.63	38
3	0.90	0.98	0.94	121
accuracy			0.71	256
macro avg	0.61	0.64	0.55	256
weighted avg	0.70	0.71	0.65	256

Figure 14: Confusion Matrix & Classification Report of InceptionV3

Figure (14) is the confusion matrix & and classification report of InceptionV3. After examining the InceptionV3 model classification report and confusion matrix, the accuracy was found to be 71.48%, which is excellent. Additionally, this research has a f1-score of around 0.71.

For EfficientNetB4:

```
Layer (type)                 Output Shape              Param #
================================================================
input_layer (InputLayer)     [(None, 256, 256, 3)]     0

sequential_1 (Sequential)    (None, 256, 256, 3)       0

efficientnetb4 (Functional   (None, 8, 8, 1792)        17673823
)

global_avg_pool_layer (Glo   (None, 1792)              0
balAveragePooling2D)

output_layer (Dense)         (None, 4)                 7172

================================================================
Total params: 17680995 (67.45 MB)
Trainable params: 7172 (28.02 KB)
Non-trainable params: 17673823 (67.42 MB)
```

Figure 15: Model Summary of EfficientNetB4.

In this figure 15 shows the model summary of EfficientNetB4. The total parameter of this model is 17,680,995. And the accuracy of this model is 86.32%. The model's accuracy and loss graph is given below.

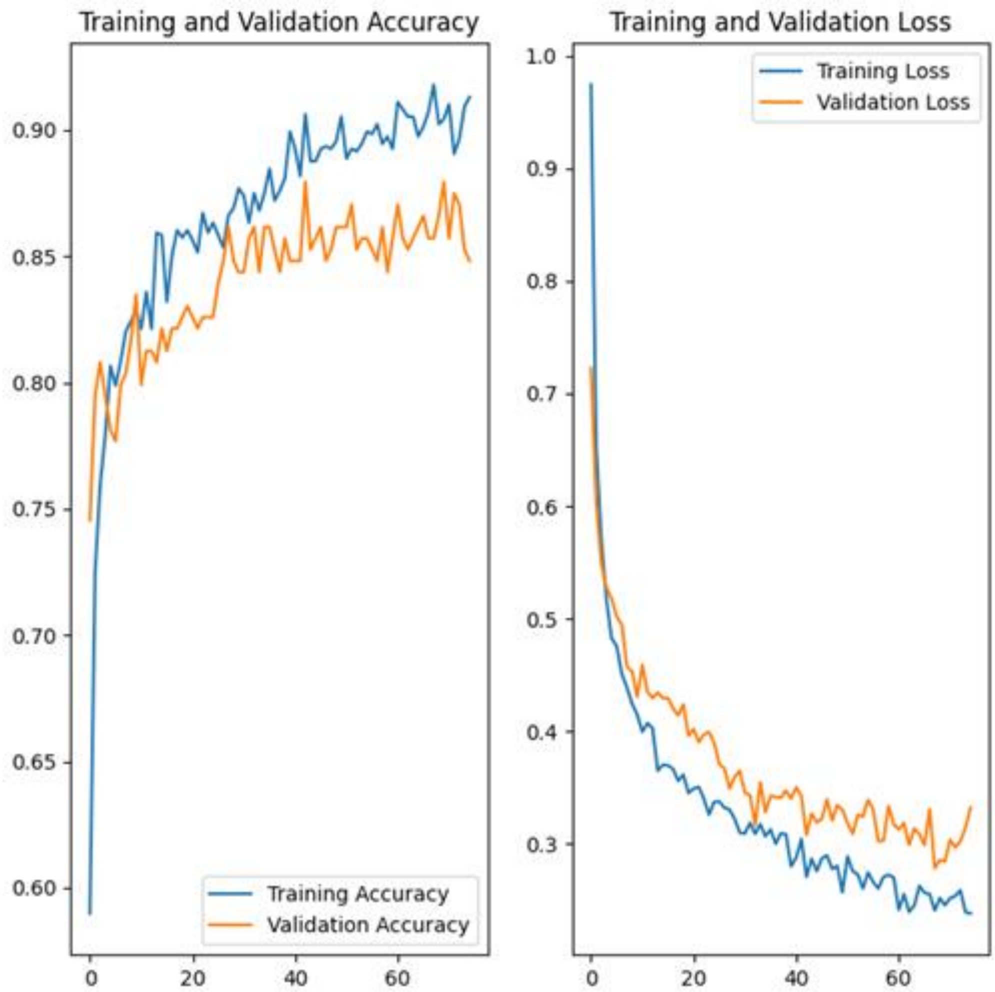
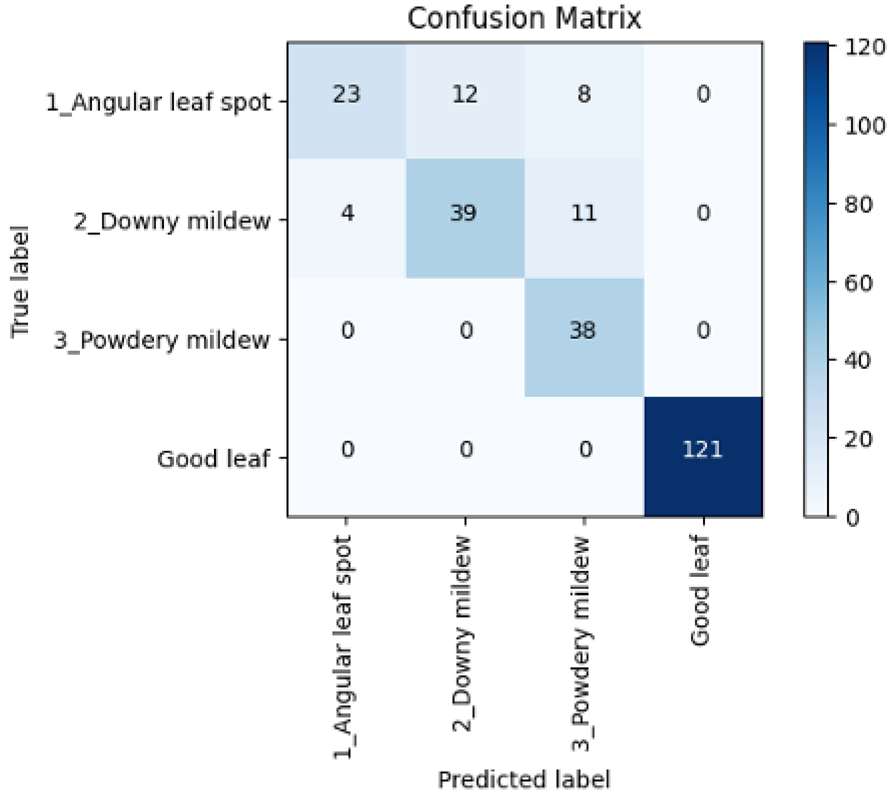


Figure 16: Accuracy and Loss Graph of EfficientNetB4.

Figure 16 is the accuracy and loss graph of EfficientNetB4. The training and validation accuracy and loss graphs in the context of the EfficientNetB4 offer important insights into how well the model is learning from the data throughout the training process and how it generalizes to new, unseen information.



	precision	recall	f1-score	support
0	0.85	0.53	0.66	43
1	0.76	0.72	0.74	54
2	0.67	1.00	0.80	38
3	1.00	1.00	1.00	121
accuracy			0.86	256
macro avg	0.82	0.81	0.80	256
weighted avg	0.88	0.86	0.86	256

Figure 17: Confusion Matrix & Classification Report of EfficientNetB4.

Figure 17 is the confusion matrix and classification report of EfficientNetB4. After examining the EfficientNetB4 model classification report and confusion matrix, the accuracy was found to be 86.32%, which is excellent. Additionally, this research has a score of around 0.86. It is essential to keep an eye on these graphs throughout training to evaluate the performance of the model, spot possible problems like overfitting, and decide whether to modify the hyperparameters or end the training to get the best results.

Table 2: Result Summary Table

Dataset	Model	Accuracy (%)	Precision (%)	Recall (%)	F1 Score
Real	CNN-1	77.34	79	77	77
Capture	CNN-2	83.59	85	84	84
Image	InceptionV3	71.48	70	71	65
Data	EfficientNetB4	86.32	88	86	86

Table 2 is the result summary table. Here we can see the best accuracy achieved by EfficienNetB4 out of other models, which is 86.32%.

Figure 18: Confidence Level of CNN Model in Predicting.

Figure 18 is the confidence level of the CNN model in predicting.

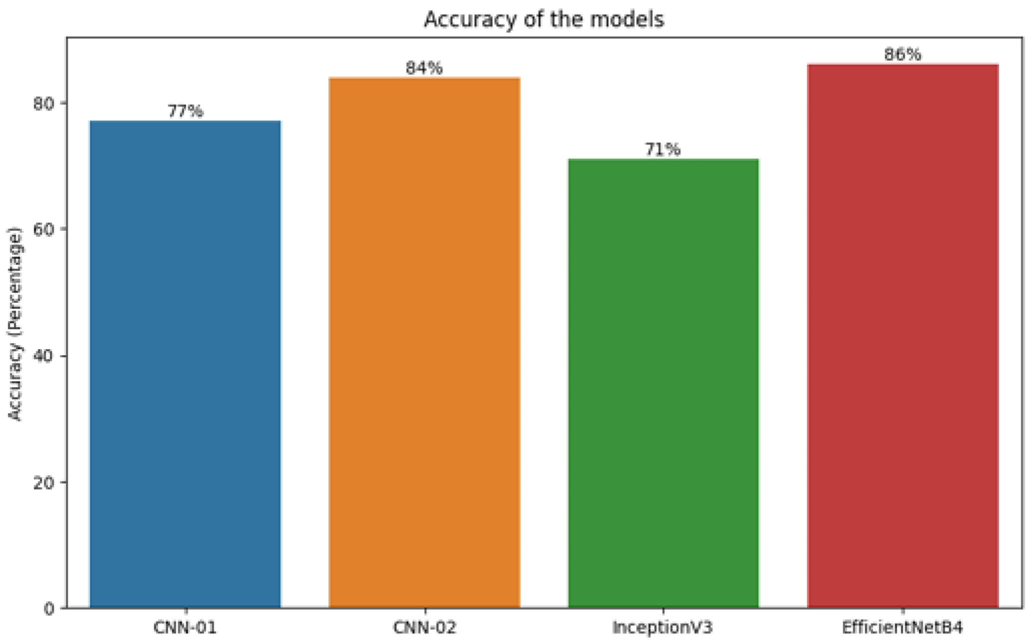


Figure 19: Percentage Line Graph.

Figure 19 is the percentage line graph. Depending on the neuron and layer number the parameter rises or falls. EfficientNetB4 gives the best accuracy of the hole work. Its percentage is 86%. The parameter determines how accurate the model is. Display the following percentage line graph now.

5. CONCLUSION

Disease control calls for an element knowledge of all elements of crop production, economics, environmental, cultural, genetics and epidemiological statistics upon which the control selections are made. Importance of Plant Diseases: Globally, giant losses of plants are because of plant diseases. Classifying vegetables based on disease classification is important for agriculture and the vegetable industry.

A lot of cucumbers are wasted every year due to untimely illness. This study implements CNN designed for image categorization, classification, and segmentation in particular. As is shown in this research, it is faster and more accurate to categorize cucumber disease classification stages using these deep learning models InceptionV3, EfficientNetB4, and Convolutional Neural Networks (CNNs) than it is to employ humans.

One of the scopes that demand more variation is multiangle images, in various light. The field of machine learning field is ever-expanding. A hybrid machine learning model approach or ensemble learning could be a scope of disease detection as well. Real field/camera captured, online resources, and dummy data are the three major plant image data. Plant leaves are self-sufficient to provide vital information, such as disease spot detection in plant disease. However, this study analyzes three diseases only. There are numerous diseases of plants that could be addressed through more thorough research as not all disease data were available in the village.

With a selected emphasis on packages in precision agriculture, this painting gives a particular approach to the automatic detection and categorization of ailments affecting cucumber leaves via the usage of deep mastering strategies. Conventional strategies for figuring out crop sicknesses regularly rely upon guide inspection, which may be laborious and subjective. By the usage of convolutional neural networks (CNNs) to investigate high-resolution pictures of cucumber leaves, the recommended deep mastering version makes it feasible to quickly it should pick out a whole lot of illnesses. As a part of the study technique, a whole lot of datasets of snapshots of cucumber leaves are gathered, consisting of samples bothered with ordinary illnesses like Angular leaf spots, powdery mildew, and downy mildew. The deep mastering version is then skilled at the dataset to permit it to find out complicated styles and capabilities related to numerous sicknesses. A distinctive series of pictures is used to validate the skilled version, displaying that it may reliably classify sicknesses in a whole lot of settings and generalize. The look demonstrates how the deep mastering version that turned into built can substantially enhance ailment control and tracking in cucumber crops. The software of precision agriculture seeks to lower the want for broad-spectrum medicinal drugs via way of means of permitting early ailment identity and well-timed action. The study's findings have ramifications that pass past the cultivation of cucumbers, as they reveal the flexibility of deep mastering inside the type of crop sicknesses across many agricultural environments.

References

[1] S. K. M. S. T. A. A. B. Md. Jueal Mia, "Cucumber disease recognition using machine learning and," *Bulletin of Electrical Engineering and Informatics,* p. 10, 2021 Dec 1.

[2] V. S. Krithika P, "Leaf disease classification on cucumber leaves using multiclass support vector," in *International Conference on Wireless Communications, Signal Processing and Networking*, 2017 Mar 22.

[3] R. Y. M. C. J. Z. L. S. Zhang J, "Identification of cucumber leaf diseases using deep learning and," *International Journal of Distributed Sensor Networks.,* 2021 Apr;17.

[4] K. M. T. U. K. S. Y. M. M. A. A. A. A. S. Hussain N, "Multiclass Cucumber Leaf Diseases Recognition Using Best Feature Selection. Computers," *Materials & Continua,* vol. 2022 .

[5] A. A. K. A. A. S. B. A. C. M. Y. H. C. J. Khan MA, "Cucumber leaf," *Applied Sciences,* 2022.

[6] Z. S. Z. C. W. X. S. Y. Zhang S, "Cucumber leaf disease identification with global pooling," *Computers and Electronics in Agriculture,* Jul 1 2019 .

[7] E. S. M. Ulutas H, "Classification of cucumber leaf diseases on images using innovative ensembles," *Journal of Electronic Imaging,* Sep 1 2023 .

[8] A.-T. J. Jasim MA, "Plant leaf diseases classification and classification using image processing," in *2020 International Conference on Computer Science and Software* , 2020 Apr 16.

[9] O. MM., "Deep learning algorithms for automatic classification and classification of mildew disease," *Fresenius Environ Bull,* Jan 1 2020 .

[10] K. M. S. M. A. T. R. A. S. T. Kianat J, "A joint framework of feature reduction," *Optik,* Aug 1 2021.

[11] W. X. Y. Z. Z. L. Zhang S, "Leaf image based cucumber disease recognition using sparse," *Computers and electronics in agriculture,* 2017 Mar 1.

[12] G. L. H. Y. L. C. P. J. Lin K, "Deep learning-based segmentation and quantification of," *Frontiers in plant science,* 2019 Feb.

[13] S. U. S. A. F. A. R. H. S. N. S. Kainat J, "Blended features classification," *Complexity,* 2021 Dec 30.

[14] D. K. Z. F. Z. L. G. Z. S. Z. Ma J, "A recognition method for cucumber diseases using leaf," *Computers and electronics in agriculture,* 2018 Nov 1.

[15] T. V. P. P. Pawar P, "Cucumber disease classification using artificial neural network. In2016," *IEEE,* 26 Aug 2016.

[16] M. R., "Cucumber Leaves Diseases Classification through Computational Approaches: A," . *Asian Journal of Research in Biosciences,* 7 Oct 2021.

[17] M. J. Asefpour Vakilian K, "An artificial neural network approach to identify fungal diseases of cucumber (Cucumis sativus L.) plants using digital image processing. Archives Of Phytopathology And Plant Protection," August 2013.

[18] T. Z. u. d. M. S. Z. A. K. M. A. S. Rehman A, "Cucumber leaf disease classification," *International Conference on Artificial Intelligence (ICAI),* 5 April 2021.

[19] S. K. F. E. U. H. K. S. I. H. Cap QH, "An end-to-end practical plant disease diagnosis system for wide-angle cucumber images," *International Journal of Engineering & Technology,* 9 October 2018.

[20] H. TANI, R. KOTANI, S. KAGIWADA, H. UGA and H. IYATOMI, "Diagnosis of multiple cucumber infections with convolutional neural networks. In2018 IEEE Applied Imagery Pattern Recognition Workshop (AIPR)," *IEEE,* 2018.

[21] D. P. W. H. L. J. Z. C. Z. H. Wang C, "A cucumber leaf disease severity classification method,"

Computers and Electronics in Agriculture., 2021.

[22] T. L. Y. N. Youwen T, "The recognition of cucumber disease based on image processing and," *Congress on image and signal processing*, vol. 2, pp. 262-267, 27 May 2008.

[23] W. Z. Jian Z, "Support vector machine for recognition of cucumber leaf diseases," *international conference on advanced computer control,* vol. Vol. 5, pp. 264-266, 27 March 2010.

[24] L. Z. W. F. W. Y. Li Y, "Hyperspectral leaf image-based cucumber disease recognition using the extended collaborative representation model," *Sensors*, 2020.

[25] G. K. A. S. Omer SM, "An Intelligent System for Cucumber Leaf Disease Diagnosis Based," *Mobile Information Systems*, 22 December 2022.