DOI: 10.15439/2022R08
ISSN 2300-5963 ACSIS, Vol. 33

Root Rot Lentil and Healthy Lentil Detection Using Image Processing

1st Noshin Tasnia
Applied statistics and Data Science
Jahangirnagar University
Dhaka, Bangladesh
nousin.tasnia@gmail.com

2nd Moon Halder
American International
University–Bangladesh
Dhaka, Bangladesh
moon.halder892@gmail.com

3rd Mst. Jiasmin Ara
Biology Department
Cantonment English School and College
Chattogram, Bangladesh
mst.jiasmin@gmail.com

4th Md. Rejaul Karim
Farm Machinery and Postharvest Process Engineering Division
Bangladesh Agricultural Research Institute
Gazipur, Bangladesh
rejaulkarim@bari.gov.bd

5th Shakik Mahmud
Computer Science and Engineering
United International University
Dhaka, Bangladesh
smahmud172174@bscse.uiu.ac.bd

***Abstract*—The hardest thing to do in agriculture is to figure out which leaves are healthy and which ones are damaged. Bangladesh makes 80% of its money from farming. Most farmers cannot read or write. They didn't know how much fertilizer to put on a lentil with root rot or a healthy lentil. They sometimes spray medicine on the plants, which is terrible for them. As a result, agriculture has become much less productive. In this paper, a picture-segmenting algorithm is given that can automatically find and classify plant leaf diseases. Also included are surveys of different ways to classify diseases that can be used to find plant leaf diseases. The Convolution Neural Network model is used to segment images, an essential part of finding plant leaf diseases. Every country's growth is based on its agricultural production. To keep agricultural production at a certain level and keep growing sustainably, scientists need to study how to find and treat diseases. Standard methods in the literature for classifying leaf images involve extracting attributes and training classifier models, which makes them less accurate. The technique suggested gets rid of any unnecessary data from the image collection. Using the mixture model for region growth, we first find the area of interest based on the colors of the leaves in the image. After extracting the features, a deep convolution neural network model is used to classify the leaf photos. A convolutional neural network model can be used with the deep learning model to find different patterns in color photos. Examining the execution strategy of the proposed model using an unauthorized dataset. According to the results of the simulating replica, the suggested model outperforms the well-known current methods in the field, with mean classification accuracy and area under the characteristics curve of 95.35% and 94.7%, respectively.**

***Index Terms*—Image processing, Root Rot lentil, Healthy lentil, CNN, Tensor flow, Plant disease detection, classification.**

I. Introduction

In today's society, the agricultural field mass is used for more than only feeding. Agriculture productivity is critical to Bangladesh's economy. As a result, in agriculture, detecting sickness in plants is vital. An automatic disease detection system helps detect plant diseases in their early stages. For example, root rot leaf disease is a dangerous disease found in pine trees in Bangladesh. The fake tree grows slowly and dies after six years. Its impact can be seen in Rajshahi, Rangpur, and Dinajpur, all of which are in the north. The early discovery could have been beneficial in such cases.The only currently used approach for identifying and detecting plant ailments is a simple naked-eye examination by experts. Working with big farms necessitates a sizable team of specialists and ongoing plant monitoring, which are expensive. However, in some nations, farmers lack the resources or skills to consult professionals. Expert consultation is, therefore, both costly and time-consuming. The recommended method is helpful in such situations for keeping an eye on vast fields of crops. Simply glancing at the signs on the plant's leaves simplifies and reduces the cost of disease diagnosis. In addition, this facilitates machine vision by enabling image-based process automation control, monitoring, and robot guidance [1] [2] [3].

It requires more time and is less accurate to identify plant diseases by eye and can only be done in certain places. However, an automated detection method takes less time and work and is more accurate. In addition to brown and yellow marks, early and late scorch, including fungus, virus, and microbial illnesses, widely planted diseases included brown and yellow spots. Image processing is used to quantify the sick area's affected area and identify the afflicted area's color difference [1] [4] [5]. By leveraging additional datasets, such as Plant Village, and CNN models trained on this database, many object identification and picture segmentation issues may be significantly improved [6]. An AI procedure called image processing separates or organizes a picture into various components. There are several different picture segmentation techniques that may be used, from straightforward thresholding to sophisticated color image segmentation. Typically, these components relate to simple things for people to distinguish and see as distinct objects. Since computers can't intelligently determine items, many different techniques for segmenting photos exist. The image processing is based on various features in the image. This could be color information, boundaries, or photo segments [7] [8]. For classified images of root rot lentils and healthy lentils, we use a convolution neural network (CNN). These papers proposed a dashboard-based system for remote data collection to support environmental parameter monitoring and provide warnings about lake water levels and hydroelectricity generation [9] [10].

II. Literature Review

Several academics have discussed several methods for detecting and identifying lentil illnesses. Revathi et al. proposed using image processing edge detection techniques to classify Cotton Leaf Spot Diseases. Initially, picture segmentation based on edge detection was performed. Hiary and Bani-Ahamed used a three-pronged approach to detect illness. The infected zone was initially detected using k-means clustering. The characteristics are then extracted using color co-occurrence methods, and ultimately, CNN is used to classify them. Arivazhagan et al. presented a four-step process for illness classification. The green pixels are masked and deleted using a threshold value in the segmentation process once a color transformation structure for the input RGB image is constructed. The texture feature is then computed using the color co-occurrence approach. The retrieved features are then sent to the classifier. The Otsu approach is used in the picture segmentation ”Leaf Spot Disease Grading Method Using Image Processing. ”ShenWei Zhang and Wu Yachun contributed. The RGB image values were transformed into the HSI color model, and the H plane value was taken for future investigation. The detection technique presented by Dhaygude and Kumbhar consists primarily of four phases. The RGB photo is converted into the HSV layer. The masking operation is carried out by mainly utilizing the values of green-colored pixels. The following functional segments are retrieved from the resulting SGDM matrices using segmentation algorithms. Camargo and J.S. Smith offer an image-processing system that analyzes colored photos to identify visual indications of plant illnesses.The study is based on an analysis of fuzzy logic, ANN, SVM, PNN, and SELF ORG MAPS. It is challenging to comprehend the algorithm’s structure and choose the best parameters for a neural network when training data is not linearly separable [4]. The objective of this research is to design a color co-occurrence method and a vision-based detection system that masks green pixels. NNs may be used to increase the recognition rate of the classification process [1]. Utilizing neural networks and K-means clustering to automatically identify leaf diseases. Artificial neural networks, fuzzy logic, and other soft computing techniques can be used to classify crop illnesses [11]. Form, color, and ideal characteristics may be employed as input criteria for sickness detection in the color co-occurrence technique utilizing SVM classifier, which allows for the expansion of training samples [2]. Improved recognition rates may be achieved by using the Gabor filter for feature extraction and the ANN classifier for classification [3]. Different plant diseases may be categorized using texture segmentation utilizing the co-occurrence matrix approach and the K-means clustering methodology, as well as the Bayes classifier, the K-means clustering technique, and the major component classifier [5]. Employing advanced color extraction features and a huge database, the color co-occurrence texture analysis technique was developed using spatial gray-level dependency matrices [12]. The picture is smoothed using the median filter, and the Otsu technique may be used to calculate the threshold. Crop loss may be estimated using the illness spot area. The condition may be classified by looking at the size of the sick spot [13]. In order to increase the accuracy of the final classification process, a study of several methods for spotting leaf disease, the creation of hybrid algorithms, and the use of neural networks have been proposed [14]. In some circumstances, the implementation still lacks the accuracy of outcomes. More tweaking is required.

III. Materials and Methodology

A. Data Preprocessing and Augmentation

Figure 1 illustrates how color picture cameras were used to record diseased blade images with a resolution of 256 and an arbitrary length from a library of leaf illnesses (Bangladesh Agriculture Research Institute) to compile three different types of disordered leaf images. Only three types of blade data were used for this investigation since some leaf illnesses might be challenging to diagnose. To confirm the identification and demonstrate the generalizability of various CNN network topologies for various leaf diseases, Apple and cherry leaves both have diseased regions, and the severity of these diseases at different disease severity levels is equal. It better satisfies the criteria for our investigation. Compared to other leaf illnesses, these sick leaves are more able to express the distinctive ability of disease areas for different CNN structures and to reach the knowledge of different CNN structures in leaf classification.

Fig. 1. Training set image of healthy lentil sample.

First, 224 × 224 resolution was achieved by resizing all images of leaf diseases to the same length and breadth. Before being fed into different networks, photos are scaled to 224 × 224 to fit the various pretraining CNN topologies. The horizontal and vertical orientations of the photos of these disease types were then reversed since certain leaf disease types contain fewer photographs than others and the gathering of leaf disease images is random.

Fig. 2. Training set of root rot lentil sample

Fig. 3. Testing set of sample image which will predict CNN model what leaf is it.

B. Convolutional Neural Network (CNN)- Based Method

Our deep learning-based network is composed of the Squeeze and Excitation (SE) module, the Inception framework, and the VGG16 convolutional layers. The first five convolutional layers were developed using the VGG16 model for self-learning low-to-high training picture features. Deeper convolutional layers extract more abstract high-level qualities by reducing the feature map resolution. The subsequent max-pooling layer cleans up the feature maps produced by the preceding convolutional layer. The feature fusion that is produced by the inception structure also increases the number of features that can be acquired from feature maps and extracts the unique features using multidimensional analysis. The greatest average pooling layer is used instead of the fully connected layer, which reduces the training parameters, speeds up convergence, and improves classification accuracy. The fully connected layer is replaced with the largest average pooling layer using the inbuilt SE module, which recalculates the special features in the channel dimension. In Fig 2 and Table 1, the modified model's network architecture is displayed together with the appropriate parameters.

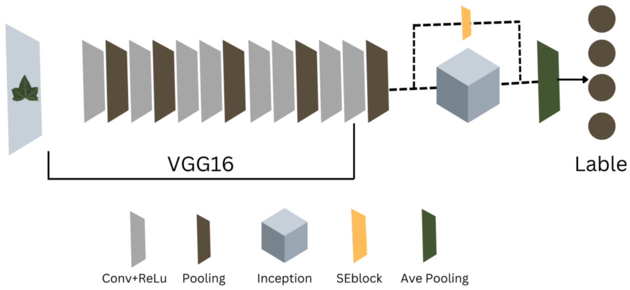

Fig. 4. The structure of the proposed convolutional neural network (CNN).

Which layers of the initial network must be frozen during the pre-training phase and which layers can continue to learn at a certain learning rate are determined by the VGG16 pre-training model. Since low-level characteristics are better able to adapt to a variety of conditions, the top few levels are often frozen. To train the model on our dataset, we used a stochastic gradient descent optimization technique. With momentum and weight attenuation set to 0.0005 and 0.9, respectively, the initial learning rate was set at 0.001. In our tests, the dropout layer was employed to avoid over-fitting training and improve the model's performance.

C. Statistical Method

The suitability of deep convolutional neural networks for the classification as described in the prior problem is assessed. We focus on two well-known designs that were created for the ImageNet dataset as part of the "Large Scale Visual Recognition Challenge" (ILSVRC): AlexNet (Krizhevsky et al., 2012) and GoogLeNet (Szegedy et al., 2015).

The LeNet-5 architecture from the 1990s's design pattern is followed by the AlexNet architecture (shown in Figure 1). (LeCun et al., 1989). Stacks of convolutional layers are often followed by one or more fully linked layers in the LeNet-5 architecture. A normalizing layer and a pooling layer may come after the convolution layers, if preferred. ReLu non-linear activation units are often used to link all network layers. Five convolutional layers, three fully linked layers, and a SoftMax layer make up AlexNet. Normalization and a pooling layer come after the first two convolution layers (conv1, 2), while only one pooling layer comes after the final convolution layer (conv5). With 38 outputs (same amount

TABLE I
CONVOLUTIONAL NEURAL NETWORK (CNN)-BASED MODEL SIMILARITY

FRAMEWORK

Type	Size	Output Size
Conv1 (Convolutional layer 1)	9 × 9/4	96 × 55 × 55
Max-pool1	3 × 3/2	96 × 27 × 27
Conv2	3 × 3/1	128 × 112 × 112
Max-pool2	3 × 3/1	128 × 56 × 56
Conv3	3 × 3/1	256 × 56 × 56
Max-pool3	3 × 3/1	256 × 28 × 28
Conv4	3 × 3/1	512 × 28 × 28
Max-pool4	3 × 3/1	512 × 14 × 14
Conv5	3 × 3/1	512 × 14 × 14
Max-pool5	3 × 3/1	512 × 7 × 7
Max-pool6	3 × 3/1	512 × 3 × 3
Inception	0	736 × 3 × 3
Max-pool7	3 × 3/2	736 × 1 × 1
Fully connection	0	256 × 1 × 1
Linear	0	10 × 1 × 1
Softmax	0	10

the total number of classes in our dataset), the last fully-connected layer (fc8) in our modified version of AlexNet feeds the SoftMax layer. The input from (fc8) is then exponentially normalized by the SoftMax layer, producing a distribution of values throughout the 38 classes that sum up to 1. These values show the network's perception of the classes that the supplied input picture belongs to. Each of AlexNet's first seven layers is connected to a ReLu nonlinearity activation unit. A dropout layer with a dropout ratio of 0.5 is connected to the first two fully connected levels (fc6, 7), in contrast. Though the network has somewhat fewer parameters (5 million parameters) than AlexNet, the GoogleNet design is more deeper and larger, with 22 layers (60 million parameters). The GoogleNet design makes extensive use of the "network in-network" architecture (Lin et al., 2013) in the form of inception modules. The inception module uses parallel 1 x 1, 3 x 3, and 5 x 5 convolutions, as well as a max-pooling layer, to simultaneously record a number of properties. The amount of labor required must be taken into account in terms of implementation practicality, which is why 1 x 1 convolutions are added for dimensionality reduction before the previously mentioned 3 x 3, 5 x 5, and also after the max-pooling layer. The outputs of all of these parallel layers are then simply concatenated in a filter concatenation layer. This is one of nine inception modules of the GoogLeNet architecture, which was employed in our research. This architecture's complete description may be found in (Szegedy et al., 2015).

In order to compare the performance of the two architectures on the Unauthorized dataset (BARI), we train the model from scratch in one case and then use transfer learning to modify already learned models (trained on the ImageNet dataset) in the other. The weights of the layer fc8 of AlexNet and the loss 1, 2, and 3 classifier layers of GooglLeNet are reset in the case of transfer learning. Then, unlike what is often done in transfer learning, we do not restrict the learning of any of the layers while training the model. To put it another way, the main difference between these two learning strategies (transfer vs. learning from scratch) is in the initial state of a few layers' weights, allowing the transfer learning strategy to take advantage of the substantial amount of visual knowledge already learned by the pre-trained AlexNet and GoogleNet models extracted from ImageNet (Russakovsky et al., 2015). Using the FDTD method to simulate wave propagation in a 2D environment, a band-pass filter (BPF) to filter out noise from the measured particle velocity of the shear wave, and then the Algebraic Helmholtz Inversion (AHI) algorithm to directly estimate CSM are the new approaches we suggest in this paper. The results of numerical simulations have demonstrated the usefulness of the recommended approach [15].

To summarize, In total of 45 experimental configurations, which depend on the following parameters:

For deep learning architecture:

- The AlexNet,
- The GoogLeNet.

Training mechanism:

- The Transfer Learning,
- The Training from Scratch.

Dataset type:

- Color,
- Grayscale,
- Leaf Segmented.

Training-testing set distribution:

- Train: 80%, Test: 20%, (In general case)
- Train: 60%, Test: 40%, (Random Case)
- Train: 50%, Test: 50%, (Random Case)
- Train: 40%, Test: 60%, (Random Case)
- Train: 20%, Test: 80%. (Rare case)

This article has made extensive use of the Architecture: Training Methodology: Datasets for Train-Test-Set-Distribution are related to certain experiments. We will use the term Google Net: 80-40 on the Transfer Learning: Grayscale to refer to the experiment employing the Google Net architecture, which was trained using transfer learning on the gray scaled BARI dataset with a train-test set distribution of 80:40. A total of 30 epochs, or the number of training iterations during which the neural network has fully traversed the whole training set, are executed for each of these 45 experiments. The decision to use 30 epochs was made based on the empirical finding that learning always converged within 30 epochs in all of these experiments (as demonstrated by the aggregated plots (Figure 2) for all of the trials).To enable a fair comparison between the outcomes of all the experimental configurations, we also attempted to standardize the hyper-parameters throughout all the experiments, using the following hyper-parameters for each experiment:

- The solver type: Stochastic Gradient Descent,
- Base learning rate is: 0.005,
- The learning rate policy: Step (decreases by a factor of 10 every 30/3 epochs)
- Momentum is : 0.9,

- The weight decay is: 0.0005,
- For gamma: 0.1,
- For batch size: 24 (in case of GoogLeNet), 100 (in case of AlexNet).

All of the aforementioned tests were conducted using our own branch of Caffe (Jia et al., 2014), a fast, open-source deep learning framework. Using a typical instance of Caffe, it is possible to reproduce the basic results, such as the overall accuracy [16].

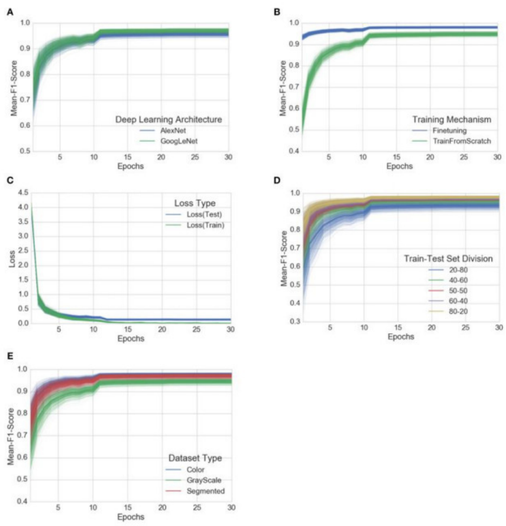

Fig. 5. The evolution of the mean F1 score and loss over the course of 30 training epochs, grouped by experimental configuration parameters.

RMSProp is chosen for our executing program for sparse reward problems.

IV. Required Technologies and Software

A. Deep Learning

Machine learning (ML), a subset of artificial intelligence, includes deep learning (DL). Artificial Intelligence is the expansive field of intelligent computer output. Machine learning focuses on developing self-learning computer algorithms that can access and utilize data. DL is a machine learning technique that uses many nonlinear transformations for data abstractions at the highest level and model designs.

B. Convolutional Neural Networks

One of the Deep Learning algorithms, a Convolutional Neural Network, can take an input image, evaluate distinct characteristics or objects in the image, and distinguish between them. Based on the properties of the design and invariance translation of their shared weights, these are also known as shift invariant or space invariant artificial neural networks (SIANN).

C. Object Detection

Object detection is a broad topic of deep learning. Object identification utilizing a convolution neural network, which is employed in many real-time areas. The deep learning neural network is used in profound learning to detect the object. Identifying items in deep learning basically involves recognizing objects in photos.

D. Tensor Flow

Tensor flow is an open source software library for high-accuracy numerical computing. Developed initially by Google Brain team researchers and engineers in the Google AI organization, it provides significant support for machine learning and in-depth learning, and the scalable digital computing base is employed in a variety of different scientific domains.

E. Keras

Keras is a Python-based high-level neural network API that can be used with TensorFlow, Theano, and CNTK. It was created with the goal of permitting. For simple experiments. Keras considers rapid and easy prototyping (through ease of use, measured quality, extensibility). Both recurrent and convolutional systems are strengthened. Runs on both the GPU and the CPU.

V. Proposed System

Digital cameras or similar technologies are used to capture photographs of various types of leaves, which are then utilized to determine the afflicted region in the leaves. Then, various image-processing techniques are used to them in order to analyse those photos and extract various and valuable information for further analysis.

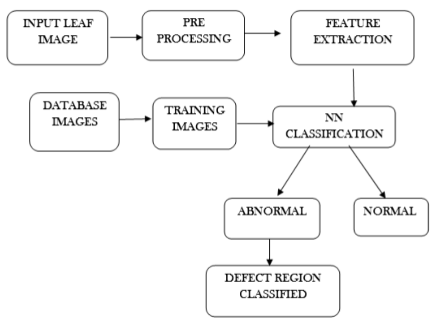

Fig. 6. Block diagram for leaf disease detection, and classification.

The following algorithm illustrates the step-by-step technique for the suggested picture recognition and segmentation processes:

- The first stage is picture acquisition, which includes capturing an image using a digital camera.
- Pre-processing the incoming image enhances image quality and removes unintended distortion. To achieve the appropriate image region, the leaf image is cropped. Then, the image is smoothed using the smoothing filter. To increase contrast, image augmentation is also employed.

- Many studies have utilized various image processing approaches to identify the leaf illness, including the following steps: picture acquisition, image preprocessing, image segmentation, feature extraction, and classification. Abirami Devaraj and colleagues [17].
- K Means clustering is used for segmentation, while CNN is utilized for classification. PCA is used in this study to decrease the feature set.
- Collect important segments for classifying leaf diseases. Using a genetic algorithm, segment the components. This paper is designed particularly for identifying leaf diseases. The work is separated into two key stages here. The ring project-based segmentation model is created in the first step to investigate the characteristics of leaf pictures.
- This paper is designed particularly for identifying leaf diseases. The work is separated into two key stages here. The ring project-based segmentation model is created in the first step to investigate the characteristics of leaf pictures.

VI. Dataset Details and Results

All the experiments are performed in Jupyter python, took 45 images for experiment. 19 images were root rot lentil and the others were healthy. Divided them into three sectors which are Training, testing and validation data set. Sample is lentil leaf images, after segmentation images will be classified which is root rot and which is healthy leaf. Firstly, read the images in our code, after then divided the training and validation dataset into two classes, then set the model CNN and library Keras layers, Maxpooling 2D, preprocessing image generator, optimizer RMSprop, activation “relu”,”sigmoid” function. To get better accuracy metrics set the loss model binary cross entropy. To model fit epochs 30. Then observation of the training model our model could predict which image is root rot and which is healthy from the validation dataset. We could use ARIMA and LSTM model for time series prediction [18].

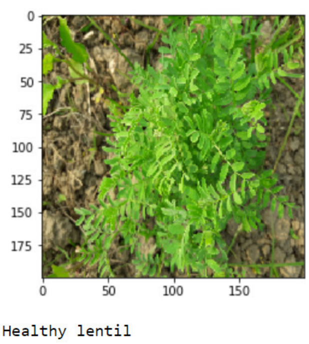

Fig. 7. Prediction output of healthy lentil leaf.

Using the best model to accurately predict the right class label (i.e., leaf and disease information) from among two probable class labels, we obtained an overall accuracy of 31.40 percent in dataset 1 and 31.69 percent in dataset 2. A random classifier will have an average accuracy of just 2.63 percent, it is important to note. In 52.89 percent of cases in dataset 1 and in 65.61 percent of cases in dataset 2, respectively, the correct class was among the top five guesses.

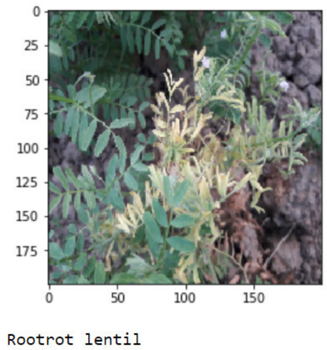

Fig. 8. Prediction of output Root Rot lentil leaf.

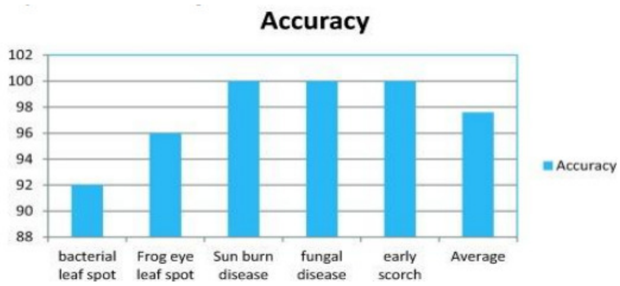

Fig. 9. Classification results per class for the proposed method.

GoogLeNet: Segmented: TransferLearning:80-20 was the best model for dataset 1 while;

GoogLeNet: Color:TransferLearning:80-20 was the best model for dataset 2.

Figures 7 and 8 show an example image from these datasets, as well as its depiction of activations in the first levels of an AlexNet architecture.

TABLE II
Plant Leaf Disease Dataset Demography

Class	Training images(before data augmentation)	Training images(after data augmentation)	Number of testing image
Leaf disease	20	18	2
Bacterial leaf spot	25	22	0
Frog eye leaf spot	33	7	1
Sun burn disease	28	14	5
Fungal disease	32	18	4
Early scorch	33	19	3

Leaf disease	Bacterial leaf spot	Frogeye leaf spot	Sun burn disease	Fungal disease	Early scorch	Average
Bacterial leaf spot	24	1	-	-	-	92
Frogeye leaf spot	23	2	-	-	-	96
Sunburn disease	-	1	24	-	-	100
Fungal disease	2	-	-	23	-	100
Early scorch	-	-	-	-	25	100
Accuracy						97.6

Fig. 10. Proposed method showing the grouping results by means of class.

TABLE III
PLANT LEAF DISEASE DATASET DEMOGRAPHY

Model	Validation Accuracy	Test Accuracy
CNN model and drop out	99.21%	99.32%
CNN model and L2 Regularization	98.62%	98.73%

Data source is collected from Bangladesh Agriculture Research Institute(BARI), the image was collected two categories one is root rot lentil leaf other is healthy lentil leaf. The data set is divided 75% for training dataset, 10% for validation and 15% for testing.

This paper outlines the key image processing techniques used for identifying leaf diseases, which include k-means clustering and CNN. This method can greatly aid in the precise diagnosis of leaf disease. Picture capture, image pre-processing, segmentation, feature extraction, and classification are the five procedures for identifying leaf diseases. By calculating the quantity of illness present in the leaf, we may use an appropriate amount of insecticides to successfully control the pests, increasing crop output. We may extend this strategy by employing other segmentation and classification algorithms.

VII. CONCLUSION

Object detection using a convolution neural network is widely employed in the emerging generation. These have a variety of applications in medicine and agriculture. The number of convolution layers can be increased or decreased to enhance accuracy. CNN validation is a type of production environment in which we may test and improve accuracy. Following the accuracy measurements in the test model and the final stage, the product implementation stage will begin. The suggested method was created with the benefits to farmers and the agricultural sector in mind. The proposed technology can identify plant illnesses and deliver disease-fighting treatments. A proper understanding of the disease and the remedy may be utilized to improve the plant's health. The suggested method is Python-based and has an accuracy of roughly 98 percent for each ailment diagnosed. Using Google's GPU for processing can improve accuracy and speed. The technology may be put on Drones to allow for aerial surveillance of agricultural areas. The project's major goal is to lessen the burden on farmers by identifying the diseases that have attacked their crops in the early stages before applying the wrong pesticide to them, causing further crop damage. Using our algorithm, we may integrate new plant disease combinations to broaden the system's scope and apply it to a variety of crops. Furthermore, the suggested technique provides a less expensive solution to reducing crop loss and educating farmers. Because Bangladesh is a country that is heavily reliant on agriculture, i.e., our agricultural exports make up a significant portion of the country's GDP, we seek to preserve the crops by presenting this model in light of the current agricultural environment in Bangladesh.

REFERENCES

[1] Dhaygude, S. B. and Kumbhar, N. P., Agricultural plant leaf disease detection using image processing. Int J Adv Res Electr Electron Instrum 3: 2(1), 2013

[2] S. Arivazhagan, R. Newlin Shebiah, S. Ananthi, S. Vishnu Varthini, Detection of unhealthy region of plant leaves and classification of plant leaf diseases using texture features. Agric Eng Int: CIGR Journal. 15(1): 211-217, 2013

[3] Kulkarni Anand H, Ashwin Patil RK. Applying image processing technique to detect plant diseases. Int J Mod EngRes. 2(5):3661–4, 2012

[4] Savita N. Ghaiwat, Parul Arora, Detection and Classification of Plant Leaf Diseases Using Image processing Techniques: A Review, IJRAET, Vol 2, Issue 3, 2014

[5] Sabah Bashir, Navdeep Sharma, Remote Area Plant Disease Detection Using Image Processing, IOSR Journal of Electronics and Communication Engineering, Volume 2, Issue 6, PP 31-34, 2012

[6] Shin HC, Roth HR, Gao M, Lu L, Xu Z, Nogues I, Yao J, Mollura D, Summers RM, Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics, and Transfer Learning. IEEE Trans Med Imaging. 35(5):1285-98, 2016

[7] B. J. Thompson, E. Dougherty, Mathematical Morphology in Image Processing (1st ed.), 1993

[8] B. Bhanu and Jing Peng, Adaptive integrated image segmentation and object recognition, in IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews),427-441,2000

[9] QT. Hoang, N.C. Minh, DT. Tran, Designing a Remote Monitoring System for Lakes Based on Internet of Things. In: DT. Tran, G. Jeon, T.D.L. Nguyen, J. Lu, TD. Xuan, Intelligent Systems and Networks. ICISN; vol 243, 2021

[10] A. S. M. A. Akib, S. Mahmud and M. F. Mridha, Future Micro Hydro Power: Generation of Hydroelectricity and IoT based Monitoring System, International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), 2021, pp. 298-302

[11] M. R. Badnakhe and P. R. Deshmukh, An Application of K-Means Clustering and Artificial Intelligence in Pattern Recognition for Crop Diseases", International Conference on Advancements in Information Technology, vol.20 (2011)

[12] Smita Naikwadi, Niket Amoda, Advances In Image Processing For Detection Of Plant Diseases, International Journal Of Application Or Innovation In Engineering and Management, 2(11), 2013

[13] B. Sanjay, Leaf disease severity measurement using image processing,in International Journal of Engineering and Technology, 3(5), 297-301, 2011

[14] P. Chaudhary, A. K. Chaudhari, A.N. Cheeran, Sharda Godara, Color Transeform Based Approach for Disease spot Detection on Plant Leaf,IJCST, 3(6), 2012

[15] T. Quang-Huy, V. T. Duong, L. Q. Hai and T. Duc-Tan, Image Reconstruction Utilizing Algebraic Helmoltz Inversion and Passband Filtering Applied to Viscoelasticity, International Conference on Multimedia Analysis and Pattern Recognition (MAPR), 1-5, 2020

[16] N. C. Minh, T. H. Dao, D. N. Tran, N. Q. Huy, N. T. Thu and D. T. Tran, Evaluation of Smartphone and Smartwatch Accelerometer Data in Activity Classification, 8th NAFOSTED Conference on Information and Computer Science (NICS), 33-38, 2021

[17] A. Devaraj, K. Rathan, S. Jaahnavi and K. Indira, Identification of Plant Disease using Image Processing Technique, International Conference on Communication and Signal Processing (ICCSP), 0749-0753, 2019

[18] N. Tasnia, S. Mahmud and M. F. Mridha, COVID-19 Future Forecasting Tool: Infected Patients Recovery and Hospitalization Trends Using Deep Learning Models, International Conference on Science and Contemporary Technologies (ICSCT), 1-6, 2021