Transfer Learning based Approach to Crops Leaf Disease Detection: A Diversion Changer in Agriculture

Md. Tarikul Islam
B.Sc. (Engg.), CSE
Begum Rokeya University, Rangpur
Bangladesh

Md. Shamsuzzaman
Assistant Professor, CSE
Begum Rokeya University, Rangpur
Bangladesh.

***Abstract*— The ever-growing agricultural sector got a pace with the adaptation of Machine Learning in its development. Plant disease detection in an early stage with the help of AI and taking protective measures from grain loss in our regular food demand is focusing researchers to find out the most effective solutions in this detection process. Computer vision and deep learning techniques can easily overcome this challenge by detecting leaf diseases more quickly than in chemical tests or human eye observation and help in pouring proper insecticides. With this moto, a multiplant leaf diseases detection (MLDD) model is proposed using deep transfer learning approach for training and identification of multi-plant leaf disease dataset with three pre-trained models MobileNet-V2, Inception-V3 and ResNet-50 to detect and identify various leaf diseases of potato, tomato and pepper, in this paper. Powerful deep features were leveraged by fine-tuning the pre-trained models. We compared and evaluate the performance of the transfer learning approach to identify and categorize crop leaf diseases. The accuracy in identifying leaf diseases was on average 97.54% in MobileNet-V2, 94.01% in Inception-V3 and 99.01% in ResNet50. Precision, recall, f1-score, and accuracy were calculated and tracked as criteria for assessing performance. This study will genuinely improve agricultural practices for food production and reduce early-stage crop loss.**

Keywords*— *Crop leaf diseases, transfer learning, MobileNetv2, ResNet-50, Inception-V3, machine learning.

I. INTRODUCTION

The world's demand for agricultural production is expanding as a result of rising population. Plant diseases in the agriculture sector result in management problems and financial losses [1]. According to the report, at least 10% of global food production is lost to plant diseases [2]. The problem is becoming more complicated as a result of the global climate change, which is also speeding up the rate of microorganisms growth and increasing the ease with which plant diseases can spread from one location to another due to the expansion of transportation system.[3]. Therefore, nowadays, early plant disease detection and management are essential in agriculture field [4]. Traditionally, plant disease detection and classification done through optical observation of plant leaves symptoms by trained and experienced human. In this method, plant disease recognition is time-consuming and error-prone. Since there are a large number of plants and their diseases contains complex symptoms, experts and rich experienced humans often fails to detect diseases which leads the diseases treatments and managements to mistake [5].

To help plant disease recognition and management many methods already have been developed. In the past decades, many lab-bases techniques have been developed to detect plant diseases. The commonly lab-based techniques are used for plant disease recognition are polymerase chain reaction (PCR), enzyme-linked immunosorbent assay (ELISA), flow cytometry, immune fluorescence (IF), DNA microarrays, and fluorescence in situ hybridization (FISH) [6]. However, to plant disease recognition, computer vision based machine learning methods have become popular where the computer becomes trained by plant disease images and then computer identify diseases.

To improve plant diseases detection accuracy, extracting the right features of the plant leaves is important [7]. Profound learning techniques, particularly CNNs, have appeared promising execution in addressing most of the challenging issues related with the classification. Deep learning techniques have improved the performance of extracting the right features which leads to improve the plant disease recognition accuracy. Although deep learning takes huge time to train a model, its testing is so fast because all information of training data becomes integrated into the deep neural network.

Our research, uses a dataset of fifteen different classes of diseases affecting potatoes, tomatoes, and peppers to train three pre-trained CNN models: MobileNet-v2, Inception-v3, and ResNet50.

The remainder section of this paper is organized as follows: Section II gives us the prior knowledge form the previous works of researchers in our working area. Section III introduces the materials and methodology of the research which entails the dataset preparation, brief overview the transfer learning models with a deep learning process flow. Section IV is all about the experimental process with model training and fine tuning. Section V discuss the experimental result and performance metrics are analyzed to achieve a better learning model and also fine tune the hyper parameter data. Section VI concludes the research work with future remarks ongoing the technological advancement in agro engineering.

II. RELATED WORKS

In recent years, Computer Vision based Machine Learning (ML) and Deep Learning (DL) are being widely used to detect plant diseases, which helps farmers identify the right plant diseases and apply the appropriate treatments. This section reviews the plant disease detection models exists in the literature.

Narmadha, Sengottaiyan, Kavitha [8] presented a deep transfer learning based rice plant disease detection model is to detect three diseases of rice (bacterial leaf blight, brown spot and leaf smut). In the model, the pertained DenseNet169 is used to feature extract. Their achieved maximum accuracy is 97.68%.

Ramesh and Vydeki [9] used a method to recognize and classify paddy leaves diseases by Deep Neural Network with Jaya Optimization Algorithm (DNN-JOA). As the part of preprocessing, the removed the background and transformed the RGB image to the HSV image. To segment the affected and healthy parts of the rice plant images they performed clustering process. The classification of the rice plant images takes by the use of DNN-JOA.

Jiang, Li, Yang, Yu [10] employed a CNN model using ResNet50 to identify tomato leaf diseases. Total 3,000 tomato leaf images are used to train their model for three classes. The average accuracy of the model has 98.00%.

Ghosal and Sarkar [11] presented a VGG16 using transfer learning method to diagnose rice plant diseases. Four classes of images are used to train this model, and average accuracy is 92.49%.

Too, Yujian, Njunki, Yingcun [12] used several pre-trained deep learning models and fine-tuned the model parameter to identify and classify different plant diseases. The model DenseNet has maximum 99.75% accuracy.

Ahmad, Hamid, Yousaf, Shah [13] used four different pre-trained deep learning models (VGG16, VGG19, ResNet, InceptionV3) to identify tomato leaf diseases. To get optimal result, they fine-tuned the model parameters. Their achieved maximum accuracy using InceptionV3 is 99.60% and 93.70% using laboratory and field plant images respectively.

Kumar, Gupta, and Madhav [14] used functional neural network, transfer learning, fuzzy logic, and convolutional neural network (CNN) with augmentation to identify diseases of coffee leaves. They have used two types of datasets (original leaf images, symptomatic portions of leaf images). Each dataset contains five classes of leaf images. Their achieved performance accuracy is 97.61%.

Ramacharan, Branowski, McCloskey, Ahmed, Legg, Hughes [15] used InceptionV3 with transfer learning to identify cassava plant diseases and pest damages (three diseases, two pest damages). The used dataset contains images with multiple number of cassava plant leaves. They have achieved higher accuracy with the dataset having a single leaf image than the dataset having images with multiple leaves.

Ramacharan, McCloskey, Baranowski, Mbilinyi, Mrisho, Ndalahwa, Legg and Hughes [16] used MobileNet model to identify cassava plant diseases with six different classes of leaf diseases. Their dataset contains both image and video files. Their achieved accuracy is 80.6% using image file and 70.4% using video files.

Chen, Zhang, Sun and Nanehkaran [17] used hybrid model using pre-trained VGG and inception layer (INC-VGGN) to identify corn and rice diseases. They made hybrid model by replacing the fully connected layers of VGGNet and they added two inception layers. Their average testing accuracy on rice diseases is 92% and 80.38% on corn diseases.

In this study, we used the Transfer Learning method to create a deep learning model using pre-trained weights taken from the well-known MobileNetv2, InceptionV3 and ResNet50 models. These additional layers in our suggested model primarily provide the lowest-cost feature extraction approach. Furthermore, fine-tuning has been done to increase the detection's precision. The dataset for our investigation comprises pictures of healthy plants as well as 15 distinct diseases. The sections below describe our system for identifying plant diseases.

III. MATERIALS AND METHOD

A. Experimental workflow

The architecture of our deep learning models is depicted in Fig. 1, which includes input datasets, pre-processing phase, deep learning model phase, transfer learning phase, classification phase, and evaluation metrics phase. First, it identify lesion patches on leaves and expose them by using suggested deep learning models. The second step is to classify diseases of leaves (Potato, Tomato, Pepper bell). To train a neural network from scratch takes a long time but transfer learning transfers the trained feature vector to train the new model on less time using less resources.

Our workflow of transfer learning-based plant disease classification of the image dataset is shown in Fig. 1. The disease images are scaled to fit into a pre-trained network’s standard input size. As the part of preprocessing step, data augmentations (rotation, scaling, shearing, zooming) are used because higher number of images make a neural network more effective. The selected pre-trained models transfer the network weights to the new model. The new features from the plant leaf images are extracted by selected network. We do not need to train the full model in transfer learning. The base pre-trained convolutional network already contains lots of features that can classifying images. However, the pre-trained model is specific for the final classification of original classification task. Unfreezing and modifying the top layers of a frozen model (transfer learning model) base and jointly train both (pre-trained model and new model) the newly-added classifier layers and the last layers of the base model.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 06, September 2022

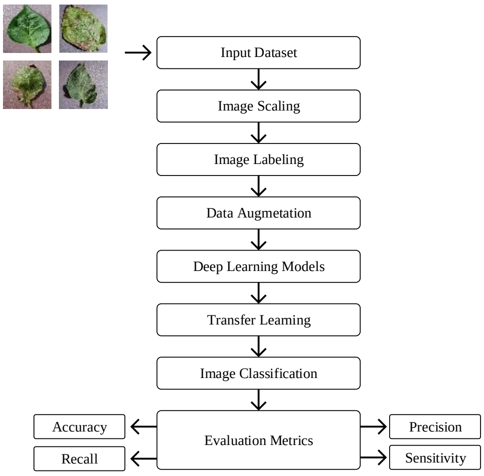


Fig. 1. Architecture of our deep learning models

B. *Transfer Learning*

A machine learning technique called transfer learning involves using pre-existing models to train new ones.Transfer learning is a methodology or method used to train a new model to perform a new task; it is not an algorithm.We just reuse the knowledge we learned during earlier training to train a new model.Transfer learning is connecting a new task in some way to an existing one.For newly trained models on fresh, unexplored data, high level generalization is necessary.

A new machine learning model can be trained without having to start from scratch thanks to transfer learning.Transfer learning can be used to train new models while saving time and resources.Although, dataset is accurately labeled, yet training takes a long time. Transfer learning, however, offers significant advantages for creating machine learning models. The key advantage is that it takes fewer resources and trains the model more effectively to train a new model. Utilizing unlabeled datasets to train machine learning models is also helpful.

The main benefits of transfer learning are:

- Not require large set of labelled training data for every new machine learning model.
- It is a generalized approach to solve machine learning problems which leverages different algorithms to solve new challenges.
- The new machine learning models can be trained within simulations rather than real-world environments.

In this work, we extracted deep features of an ImageNet weight-based, pre-trained model as shown in Fig.2. Here, deep features, refer to features extracted from the DCNN. For this, we used three pre-trained models (MobileNet V2, Inception V3, ResNet50). Pre-trained models are trained on ImageNet Dataset [18], which contains 1000 classes. The weights were originally obtained by training on the ILSVRC-2012-CLS dataset for image classification ("Imagenet").

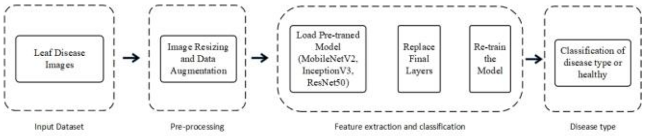


Fig. 2. Transfer learning workflow

C. *Experimental models*

1. *MobileNetV2:*

MobileNetV2 is a convolutional neural network architecture designed for mobile devices. It is built on an inverted residual structure with residual connections between bottleneck layers. As a source of non-linearity, the intermediate expansion layer filters features using lightweight depth wise convolutions. The MobileNetV2 architecture includes an initial fully convolution layer with 32 filters, followed by 19 residual bottleneck layers. [18]. With the limited memory of Mobile devices MobileNetv2 is a better option for classification tasks as, this Network requires small number of parameters compared to other networks. MobileNetv1 used 4.2millions parameters whereas in MobileNetv2 it is reduced to 3.2million parameters and with better accuracy.

2. *Inception-V3:*

Inception-v3 is another convolutional neural network architecture from the Inception family (GoogLeNet or Inception-v1, Inception-v2) that makes several improvements including using Label Smoothing, smaller and asymmetric convolution, Factorized 7 x 7 convolutions, and the use of an auxiliary classifier to propagate label information lower down the network overhead along with the use of batch normalization for layers in the sidehead [19]. The auxiliary classifier tackles the problem of vanishing gradient. Inception-v3 was experimented with ILSVRC12 for the classification task of ImageNet 1000 classes.

Factorization of the convolutions layer into smaller layers and the grid size reductions inside the InceptionV3 network results in relatively low computational cost with high quality classification task. With the utilization of fewer number of parameters count and additional batch-normalized auxiliary classifiers regularization process and label smoothing allows the training of the network with higher quality while the number of train datasets is not enough in number.

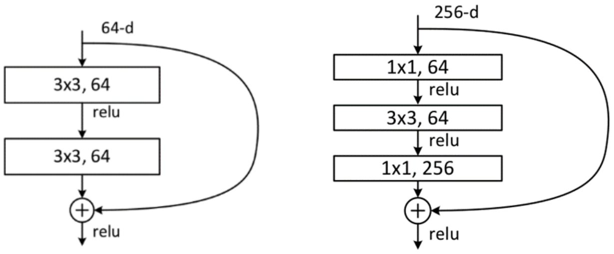
3. *ResNet-50:*

Deep residual networks like the popular ResNet-50 model is a convolutional neural network (CNN) that is 50 layers deep (48 Convolution layers, 1 MaxPool and 1 Average Pool layer). It has 3.8 x 10^9 Floating points operations. A Residual Neural Network (ResNet) is an Artificial Neural Network (ANN) of a kind that stacks residual blocks on top of each other to form a network [20].

The model is tested with ImageNet 2012 dataset consisting of 1000 classes of images (1.28 million images for model training and 50k images for model validation) and network performance is tested with 100k images. It has 3.8blilions of FLOPS. ResNet-50 is replaced with 3 layers of bottleneck blocks than 2 blocks in prior model, as we can see in Fig. 3. Among 50 layers of this model, Layer-1 started with 64

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 06, September 2022

convolution kernel with the size of 7*7 and stride 2. By varying the kernel size and number of layers are grouped in different identical block as, 1 + 9 + 12 + 18 + 9 + 1 = 50 layers and the last layer is a fully connected layer with 1000 nodes and used the softmax activation function for classification task. Thus, this model resolved the vanishing gradient problem in learning very deep neural networks.



(a) 2-bottleneck layers (b) 3-bottleneck layers

Fig. 3. ResNet-50 bottleneck layers

4. Dataset:

The dataset used in this work was taken from the PlantVillage Kaggle competition, which was supported by Arjun Tejaswi [21]. The dataset consists of 20,637 squared images and their dimension is 256x256 pixels. Dataset consists fifteen diseases of Potato, Tomato and Pepper. We have divided the dataset into train (70%), validation (20%), test (10%), and with batch size 32. In our dataset, Pepper bell has two classes: one is for leaf disease bacterial spot and another is healthy leaf containing 997 and 1478 numbers of images respectively, as shown in Fig. 4. There are 2152 numbers of potato leaf images representing three different classes: early blight, healthy, late blight, shown in Fig. 5. Tomato leaf disease dataset contains ten classes: target spot, mosaic virus, yellow leaf curl virus, bacterial spot, early blight, late blight, healthy, leaf mold, septoria leaf spot, spider mites of data with number of 16010 images respectively as shown in Fig. 6. Images were rescaled into 224×224 in resolution to meet the input feature vector requirement of our prospected experimental CNN models. We labeled our dataset according to the name of plant, tagged with the disease name, so that the multiclass disease data can be identify and classified properly.

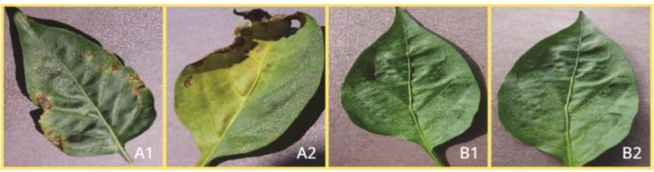


Fig. 4. Sample pepper bell plant leaf image (A1, A2: bacterial spot; B1, B2: healthy)

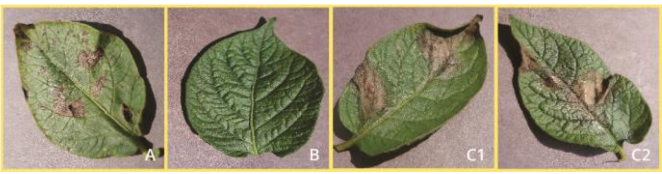


Fig. 5. Sample potato plant leaf image (A: early blight; B: healthy; C1, C2: late blight)

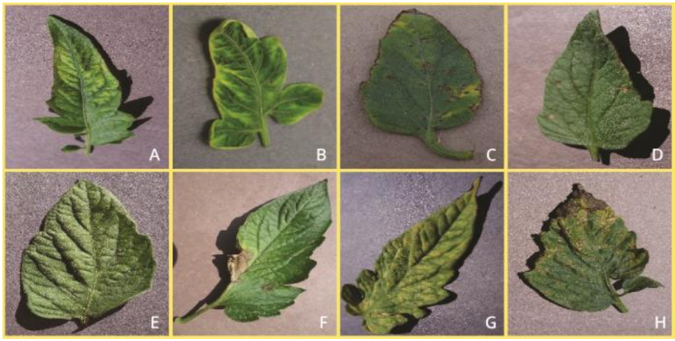


Fig. 6. Sample tomato plant leaf image (A: target spot; B: mosaic virus; Cd: yellow leaf curl virus; D: bacterial spot; E: healthy; F: late blight; G: leaf mold; H: septoria leaf spot)

IV. EXPERIMENT

A. Experimental Setup

All models used in this work were compiled with graphical processing-unit (GPU) support. The experiment was performed using Python v3.9 on a laptop running Windows 11 with Nvidea GeForce MX110 with 8GB memory. Processor was Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 1.80 GHz. Tensorflow, tensorflow hub, Keras, PIL, numpy, OpenCV, Matplotlib libraries were used in the Anaconda Jupyter Notebook.

B. Experimental Process

Pre-train models are used from tensorflow hub. For the models (MobileNetV2, InceptionV3, ResNet50) we set 10 epochs, batch size is set to 32 so that we can deal with the data overfitting issue. The number of training set steps to the total number of training sets divided by the batch size (32). The model weights and all the deviation parameters become updated. Similar to the training set, total number of the verification sets divided by the batch size (32). we calculate the accuracy and loss function in the back propagation process for both training and validation process. We set the learning rate to 0.001 (default for training process) where the reasonable setting of the learning rate can reduce the gradient problem of the loss function between the model output and the expected value, which can ensure reasonable training.

C. Model Training and validation

The training process is carried out through modification of MobileNetV2, Inceptionv3 and ResNet50 by freezing the lower layers and then trains it with the training set of images. All layers were frozen except the las two fully connected layers of the models and fine-tuned to adjust the models . With batch size of 32 total number batches were 452 for training batches and 156 validation batches.

Training and validation accuracy plot and training and validation loss plot of MobileNetV2 in Fig.6 describes the training and validation improvement by epochs. The blue curve indicates training performance and orange one is for identifying validation performance. According to our graph, accuracy is in increasing fashion and loss in decreasing trends. We cut off the process while it reaches at pick level and tend to decrease in case of accuracy and cut off the falling loss function value at its rising stage. In case of Accuracy the cutoff point was 97.54% for training and 92.52% for validation in epoch 10.

Training and validation accuracy plot and training and validation loss plot of InceptionV3 in Fig.7 describe the training and validation improvement by 10 epochs. Accuracy curve is increasing where validation curve decreasing in every epochs. Last epoch gives 94.01% training accuracy, 86.09% validation accuracy. Training and validation accuracy plot and training and validation loss plot of ResNet50 in Fig.8 describe the training and validation improvement by 10 epochs. Accuracy curve is increasing where validation curve decreasing in every epochs. Last epoch gives 99.01% training accuracy, 93.17% validation accuracy. Among three of the trained models, ResNet-50 has minimum loss with the highest accuracy for both in training and validation process, as shown in Table I. Thus, ResNet-50 can be a better training model for researchers in identifying and classifying crops leaf diseases.

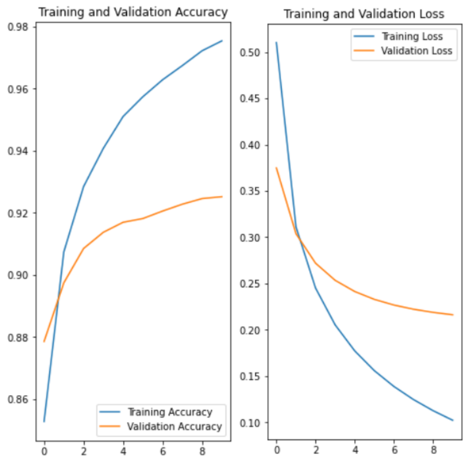


Fig. 6. MobileNetV2 Training and validation

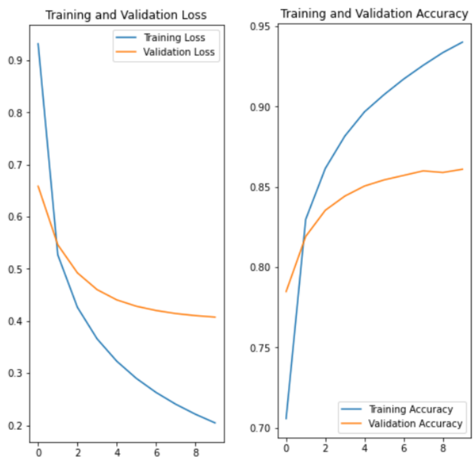


Fig. 7. InceptionV3 Training and validation

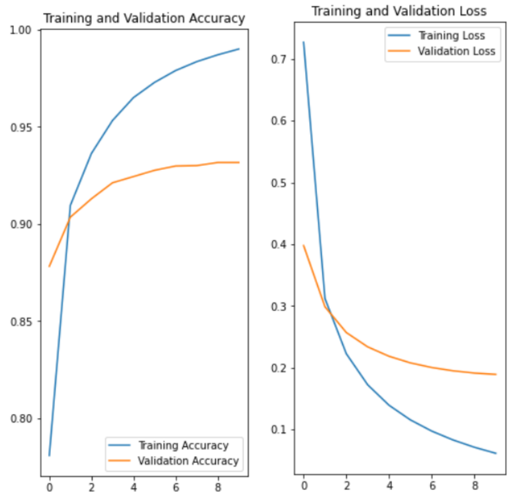


Fig. 8. ResNet50 Training and validation

TABLE I. ACCURACY AND LOSS IN TRAINING AND VALIDATION OF MODELS

Training Models	Accuracy (%)		Loss (%)	
	Training	*validation*	*Training*	*validation*
MobileNetv2	97.49	92.36	10.31	21.38
Inception-V3	94.01	86.09	20.46	40.74
ResNet-50	99.01	93.17	6.11	18.89

V. RESULT AND DISCUSSION

Pretrained CNN models are trained and validated with our dataset adopting their predefined weighted values. With batch size of 32 there were 38 batches for test case and got a class wise identification measures through a statistical calculation of performance evaluation parameters in Table II. A confusion matrix is presented from which we can get better understanding of the class-wise predictions of the disease as shown in figure 5.2 and table 5.2.

A. Perfromance Evaluation

Precision, Recall and F1 score are performance evaluation metrics used to identify the overall accuracy of training models for some datasets. Hence, precision measures the number of positive predictions that are correct (TP). Recall (also called sensitivity) measures the number of positive cases that a model is able to predict correctly among all positive cases in dataset. F1 score tells us the harmonic mean on precision and recall. It has some advantages over precision and recall. In case of very small values of precision and recall, it balances the metrics.

$$Precision=TP/(TP+FP) \quad (1)$$
$$Recall=TP/(TP+FN) \quad (2)$$
$$Accuracy= (TP+TN)/(TP+FN+TN+FP) \quad (3)$$
$$F1=(2\times Precision\times Recall)/(Precision+Recall) \quad (4)$$

Here, true positive rate is represented by TP. True positive or TP is the total number of garbage detected correctly. False positive rate is represented by FP. False positive or FP is the false predicted garbage value from total garbage value. False negative rate is represented by FN. False negative or FN is the garbage predicted value was falsely predicted. The harmonic mean of recall and precision is used to calculate the

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 11 Issue 06, September 2022

F measure, giving each the same weight. It enables evaluation of a model taking into consideration both precision and recall using a single score, which is useful for explaining the performance of the model and when comparing models. For example, performance measure in classifying pepper bell bacterial spot with MobileNetV2 is as follows:

TP=69, FP=1, FN=5, TN=1137
Precision = 69/(69+1) = 0.9857
Recall = 69/(69+5) = 0.9332
Accuracy = (69+1137)/1212 = 0.9950
F1-Score= (2×0.9857×0.9332)/(0.9857+0.9332) =0.958

Table II represents the average values of accuracy, precision, recall and f1-score which are calculated from class-wise values of corresponding models. We can see that, the model ResNet50 has higher accuracy which is 99.01% in predicting a true class other than our experimented two models, ModelNetv2 and Inceptionv3 and also in precision, recall and F1 score ResNet50 achieved more than 90% above performance. Fig. 8. demonstrates the overall performance of our three experimented CNN architectures and from the illustration it is quite clear that for the classification task of our plant disease dataset, ResNet50 is better choice.

A confusion matrix is the most basic and intuitive method to evaluate a classification model. In image classification, the confusion matrix allows the comparison of the classification with the actual measurement values. The ability of the tested models to correctly identify 15 types of plant leaf disease was measured using confusion matrices. We have also calculated the confusion matrix to evaluate the performance of the models as show in Fig. 9. As it is visible from confusion matrix heat map representation, highest prediction is 190 which is in ResNet50 model, whereas in MobileNetV2 and InceptionV3, 184 and 176 respectively. All true positive result for all other classes in three models of classification task in confusion matrix, shows a high precision for ResNet50. Some images are misclassified, this may be because of very close features similarity among few disease such as in early blight and late blight and incase of bacterial spot also. More samples of images can be a solution for this overlapping. A class-wise performance comparison for tested 15 different classes of leaf disease is shown in Table III. We can observe now, for which category of disease our machine learning models have a better performance. For bacterial spot, mosaic viruses, early blight and late blight of potato the models have lower precision, recall and f1 measure which is below 85%. Bacterial spot, septorial leaf spot, tomato's early and late blight models have higher precision, recall and f1 measure which is 95% above. Among all three models ResNet50 has superior performance for all classes (varying from 90% – 100%)

TABLE II. PERFORMANCE METRICS OF TRAINING MODELS (MOBILENETV2, INCEPTIONV3, RESNET50)

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
MobileNetV2	97.49	93.00	91.00	93.00
InceptionV3	94.01	89.00	87.00	86.00
ResNet50	99.01	94.00	94.00	94.00

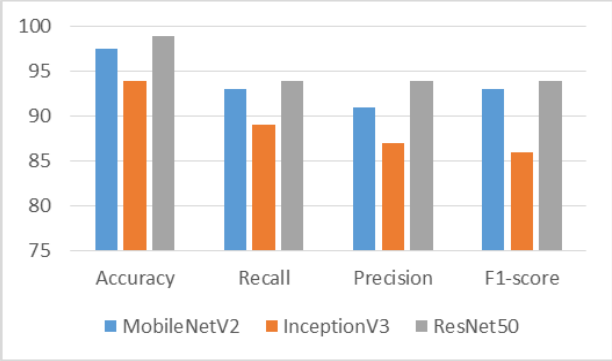
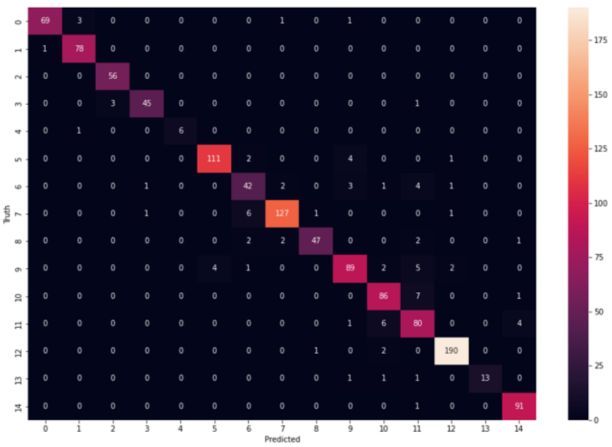
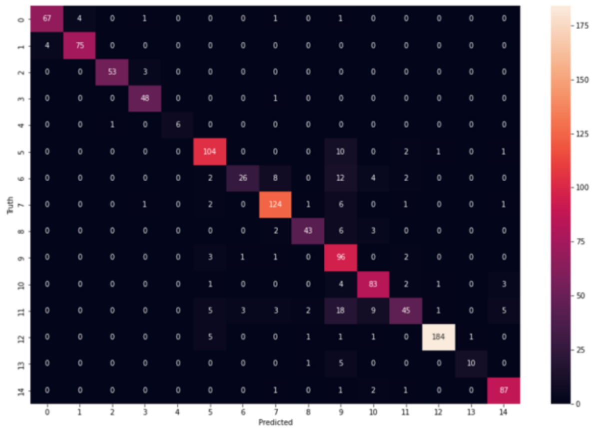


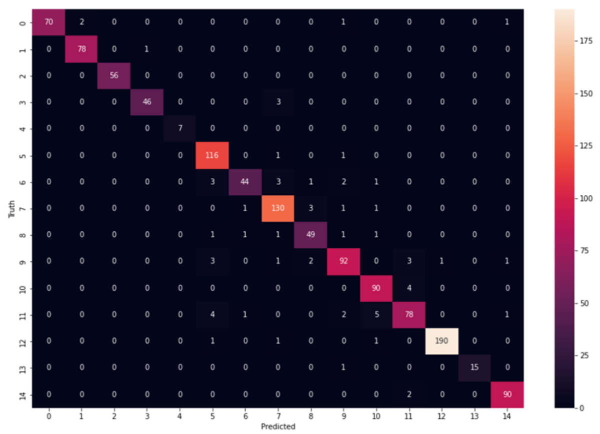
Fig. 8. Performance illustration of three models (MobileNetV2, InceptionV3, ResNet50)



(a)



(b)



(c)

Fig. 9. Confusion matrix of (a) MobileNetV2, (b) InceptionV3 and (c) ResNet50

VI. CONCLUSION

Plant diseases have a catastrophic effect on the safety of food production and are the primary threats to global agricultural development. In extreme circumstances, plant diseases could altogether prevent a harvest. Therefore, it is particularly desirable in agricultural information for plant diseases to be automatically identified. The majority of the difficult classification problems have been successfully addressed by deep learning techniques, particularly CNNs.

In order to improve the earning potential of micro lesion symptoms, transfer learning for deep CNNs is examined in this study. The effectiveness of the model has been evaluated through comparison with a few different transfer learning models and appropriate graphs. Our experimental models are capable of differentiating between 15 types of both disease-free and unhealthy leaves. Experimental findings showed that the ResNet50 model performed at a cutting-edge level. Training the network requires much less time as compared to the standard CNNs. The training and validation accuracy for ResNet50 99.01% and 93.17%. The outcome of the experiment reveals that the suggested model obtains a greater level of efficiency and reliability. MobileNetV2, InceptionV3, and ResNet50 all provide average accuracy in disease identification of 97.54%, 94.01%, and 99.01%, respectively. ResNet50 outperformed these models in terms of performance.

TABLE III. CLASS WISE PERFORMANCE MEASURES

Plant	Disease	MobileNetv2			Inceptionv3			ResNet50		
		Precision (%)	Recall (%)	F1-score (%)	Precision (%)	Recall (%)	F1-score (%)	Precision (%)	Recall (%)	F1-score (%)
Pepper Bell	Bacterial spot	99	93	96	93	90	92	100	96	98
	Healthy	95	99	97	94	94	94	97	99	98
Potato	Early blight	95	100	97	98	95	97	100	97	98
	Healthy	96	92	94	89	94	92	98	91	94
	Late blight	100	86	92	100	88	93	89	100	94
Tomato	Target spot	97	94	95	86	87	87	92	99	95
	Mosaic virus	78	79	78	87	47	61	92	82	87
	Yellow leaf curl virus	96	93	95	88	93	90	91	95	93
	Bacterial spot	96	87	91	89	75	82	88	87	88
	Early blight	90	86	88	61	94	74	90	88	89
	Healthy	88	91	90	85	91	88	89	96	92
	Late blight	79	88	83	84	51	63	90	84	87
	Leaf mold	97	98	98	98	96	97	99	98	99
	Septoria leaf spot	100	81	90	91	71	80	93	100	97
	Spider mites	94	99	96	91	95	92	97	99	98

The accuracy wasn't 100 percent accurate for all classes. It's presumably because of how the leaves are. Some species' pictures are quite similar to one another in terms of texture, color, and shape. As a result, it can be exceedingly difficult for networks to anticipate the true labels accurately.

In the future, we will continue to examine how well plant disease identification performs using more reliable datasets and will offer a cutting-edge approach to achieve greater performance. We intend to use it for more practical purposes. A program can be created and made available for farmers' mobile devices. Farmers will be able to decide for themselves when using an application that will identify plant diseases and recommend fertilizer and insecticides to use.

REFERENCES

[1] S. S. Abu-Naser, K. A. Kashkash and M. Fayyad, "Developing an expert system for plant disease diagnosis," Journal of Artificial Intelligence, vol. 1, no. 2, p. 269–276, 2010.

[2] R. N. Strange and P. R. Scott, "Plant disease: a threat to global food security," Annual Review of Phytopathology, vol. 43, pp. 83-116, 2005.

[3] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," Comput. Intell. Neurosci, 2016.

[4] J. G. A. Barbedo, "Factors influencing the use of deep learning for plant disease recognition," Biosystems Engineering, vol. 172, pp. 84-91, August 2018.

[5] K. P.Ferentinos, "Deep learning models for plant disease detection and diagnosis," Computers and Electronics in Agriculture, vol. 145, pp. 311-318, February 2018.
[6] S. Sankaran, AshishMishra, R. Ehsani and C. Davis, "A review of advanced techniques for detecting plant diseases," Computers and Electronics in Agriculture, vol. 72, pp. 1-13, June 2010.
[7] Y. G. Naresh and H. S. Nagendraswamy, "Classification of medicinal plants: An approach using modified LBP with symbolic representation," Neurocomputing, vol. 173, pp. 1789-1797, January 2016.
[8] R. P. Narmadha, N. Sengottaiyan and R. J. Kavitha, "Deep transfer learning based rice plant," Intelligent Automation & Soft Computing, July 2021.
[9] S. Ramesh and D. Vydeki, "Recognition and classification of paddy leaf diseases using," Information Processing in Agriculture, vol. 7, no. 2, pp. 249-260, 2020.
[10] D. Jiang, F. Li, Y. Yang and S. Yu, "A tomato leaf diseases classification method based on," in Chinese Control And Decision Conference (CCDC), 2020.
[11] S. Ghosal and K. Sarkar, "Rice leaf diseases classification using CNN with transfer," in IEEE Calcutta Conference (CALCON), Kolkata, 2020.
[12] E. C. Too, L. Yujian, S. Njuki and L. Yingchun, "A comparative study of fine-tuning deep learning models for plant disease identification," Computers and Electronics in Agriculture, vol. 161, pp. 272-279, June 2019.
[13] I. Ahmad, M. Hamid, S. Yousaf, S. T. Shah and M. O. Ahmad, "Optimizing pretrained convolutional neural networks for tomato leaf disease detection," Complexity, pp. 1-6, September 2020.
[14] M. Kumar, P. Gupta and P. Madhav, "Disease detection in coffee plants using convolutional," in International Conference on Communication and Electronics Systems (ICCES), Coimbatore, 2020.
[15] A. Ramcharan, K. Branowski, P. McCloskey, B. Ahmed, J. Legg and D. P. Hughes, "Deep learning for images-based cassava disease detection," Front. Plant Sci, vol. 8, pp. 1-1852, October 2017.
[16] A. Ramcharan, P. McCloskey, K. Baranowski, N. Mbilinyi, L. Mrisho, M. Ndalahwa and J., "A mobile-based deep learning model for cassava disease diagnosis," Front. Plant Sci, vol. 10, pp. 1-272, March 2019.
[17] J. Chen, D. Zhang, Y. Sun and U. A. Nanehkaran, "Using deep transfer learning for image-based plant disease identification," Computers and Electronics in Agriculture, vol. 173, 2020.
[18] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L. C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2018.
[19] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, 2016.
[20] X. Z. S. R. J. S. K. He, "Deep Residual Learning for Image," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
[21] [21] A. Tejaswi, "Plant Village," Kaggle, 2019. [Online]. Available: https://www.kaggle.com/datasets/arjuntejaswi/plant-village. [Accessed 12 February 2022].