Tomato leaves Diseases Classification using Deep Learning Architecture

[1]Nirmala M. S., [2]Prem Singh M.

[1]Assistant Professor. Government College for Women (Autonomous)

[2]HOD and PG Coordinator Government College (Autonomous), Mandya

Article Info
Page Number: 333 - 350
Publication Issue:
Vol 72 No. 2 (2023)

Abstract: Disease in the plants can significantly reduce the quality and quantity of the plant yield. A disease is quite natural in plants, which is why disease detection in plants is so important in farming. Diseases are responsible for both direct and indirect mon etary loss to the farmer. Classification of plant disease is imperative, which helps in enchanting a proper management step towards preventing the further spread of the disease once the disease is correctly classified and treated suitably. Deep learning methods have demonstrated significant improvements in plant leaf classification performance. The rapid and precise diagnosis of disease severity will aid in reducing yield losses. The work presented here focuses on the classification of Tomato plant leaves disease. The nine classes of healthy and diseases of tomato plant leaves from the PlantVillage dataset that occurs in Indian states are studied here. The diseases of tomato plant leaf analyzed here are Bacterial Spot (BS), Early Blight (EB), Late Blight (LB), Leaf Mold (LM), Mosaic Virus (MV), Septoria Leaf Spot (SLS), Target Spot (TS), and Yellow Leaf Curl Virus (YLCV).

Article History:
Article Received: 15 July 2023
Revised: 24 August 2023
Accepted: 18 July 2023

Keywords: - Alexnet, GoogleNet, AlexNet, SqueezeNet VGG16, GoogLeNet, MobileNet, ResNet-18, ResNet-50, ResNet-101

Introduction

In Agriculture its highly imperative to diagnose of the plant disease in an early stage. Detection of plant disease through leaves is a generally used technique to figure out the disease as it shows the change in its original structure for different diseases. Identifying the disease by the naked eye of an expert needs vast professional experience with extensive knowledge about the causes of disease on the crops (Ma et al. [1]). Furthermore, the expert should have sufficient knowledge to inform the details related to signs and symptoms caused by disease. Even today, manual evaluation is done in remote villages, but it does not identify the exact disease and its variants. Manual assessment is a time-consuming process for larger farms and requires huge manpower. Moreover, cultivation is a continuous process, thus needs periodic monitoring of crops to find out the disease. As a result, an alternative method is required to identify the diseases automatically utilizing leaf images (Barbedo [2]). Recently, advances in computer vision present an opportunity to expand and enhance the practice of precise plant protection and extend the market of computer vision applications in the field of agriculture. In traditional machine learning techniques, most of the applied features need to be identified by a domain expert in order to reduce the complexity of the data and make patterns more visible to learning algorithms to work. Many Machine Learning (ML) models have been employed for the detection and classification of plant diseases. The advancements in a subset of ML, that is, Deep Learning (DL) utilizes a hierarchical level of artificial neural networks to carry out the process of machine learning. In deep learning, a computer model learns to perform classification tasks directly from images. Deep learning models are trained by using large set of labelled data and neural network architectures that learn features directly from the data without the need for manual feature extraction. Deep Convolution Neural Network (DCNN) is a DL method that consists of many neural network layers. DCNN models assume that with the increase in depth, the network can better approximate the target function with a number of nonlinear mappings and more enriched feature hierarchies (Bengio [3]). The powerful learning ability of DCNN is primarily due to the use of multiple feature extraction stages that can automatically learn representations from the data. The multi-layered, hierarchical structure of DCNN, gives it the ability to extract low, mid, and high-level features of the tomato leaves. High-level features (more abstract features) are a combination of lower and mid-level features. DCNN differs from normal Convolution Neural Network in terms of number of hidden layers which are used to extract more features and increase the accuracy of the prediction. There are two kinds of DCNN, one is increasing the number of hidden layers and another by increasing the number of nodes in the hidden layer.

1. Related work

Basavaiah & Anthony [4] presented multiple features fusion method which identifies four main leaf disease of tomato such as bacterial spot, septoria spot, yellow curl virus, and mosaic virus. The features such ashu moments, colour histogram, local binary pattern, and haralick are extracted from the leaves. Subsequently extracted features are used by the random forest and decision tree algorithms to perform classification. The experimental result confirms that random forest shows the highest detection accuracy of 94% whereas the decision tree is 90%. Annabel & Muthulakshmi [5] reported detection of three typesof tomato leaf disease, including bacterial spot, tomato mosaic virus, late blight and healthy leaf through a Random Forest (RF) algorithm. The algorithm achieved an accuracy of 94.10% which is highest compared to SVM which is of 82.60% and MDC 87.60% obtained on the same dataset. Foysal *et al*. [6] reported a novel CNN model that consists of 15 layers to identify the occurrence of five different tomato diseases using leaf images. The experimental result shows that the proposed model achieved an accuracy of about 76% on the test dataset, which consists of 600 images. Ruedeeniraman *et al* [7] developed an embedded based VegeCare tool using a deep CNN model, which identifies six varieties oftomato leaf diseases. Sardogan *et al.* [8] proposed a Learning Vector Quantization (LVQ) based deep CNN model on classifying four types of tomato disease and a healthy class. The proposed method achieved an 86% of classification accuracy. Rubanga *et al.* [9] detected the infestation of the tutaabsoluta virus in the tomato leaves using pre-trained CNN architecture such as Inception V3, VGG16, VGG19 and ResNet. The result obtained from the experiment shows that Inception V3 produced the highest detection accuracyof 87.20% than VGG16, VGG19 and ResNet. Chen *et al.* [10] developed a framework combining Artificial Bee Colony algorithm (ABCK), Binary Wavelet

Transform combined with Retinex (BWTR), and Both-channel Residual Attention Network model (B- ARNet) to recognize five different types of tomato leaf diseases such as early blight, late blight,citrinitas leaf curl, leaf mold, and bacterial leaf spot. The experiment is conducted using 8616 images of tomato leaves obtained fromthe Hunan Vegetable Institute. The proposed method based on the combination of ABCK-BWTR and B-ARNet achieved a classification accuracy of 89%.

Agarwal *et al*. [11] developed a CNN model for disease identification of tomato plants using leaf images. The proposed model uses 3 convolution layers, 3 max-pooling layers, and 2 fully connected layers to classify 9 diseased class and a healthy class. The experimental output signifies that the proposed model achieves the highest test accuracy of 91.20% compared with pre-trained models such as VGG16-77.20%, MobileNet- 63.75%, and Inception V3-63.40%. Liu & Wang [12] build an improved YOLO V3 model to identify tomato diseases and pests in a real-time environment. The proposed model employs image pyramid to perform multi-scale feature detection in order to enhance the detection accuracy and speed of the model. Subsequently, the model can accurately detect the region and type of the diseases and pests spotted in the tomato plant. Based on the experimental result, improved YOLO V3 model achieved the highest average recognition accuracy of 92.36% compared with models such as SSD, Faster R-CNN and original YOLO V3 models. Yet another study attempted by, Liu & Wang [13] developed the YOLO V3-MobileNetV2 model to recognize the tomato Gray leaf spot in the early stage. The experimental result showed that the proposed model provides high recognition accuracy for early disease prediction compared with the Faster-RCNN and SSD models. Gadekallu *et al*. [14] proposed a deep learning model using Principal Component Analysis (PCA)-Whale Optimization Algorithm (WOA) and Deep Neural Network (DNN) to classify tomato leaf diseases. The tomato leaf images employed for the are collected from the PlantVillage database, which consists of nine types of diseased class and a healthy class. The PCA-WOA approach is used to identify the significant features of the tomato leaf and is fed to the DNN which performs the classification. The proposed model achieves a test accuracy of 94%. Prabhakar *et al*.[15], developed an intelligent system using foldscope and ResNet101 to classify the severity level like mild, moderate,and severe of the early blight disease in the tomato leaf. The ResNet101 achieves an accuracy of 94.6% in severity assessment of disease which is highest as compared which the other CNN models such as VGG16, VGG19, AlexNet, GoogleNet, and ResNet101. Karthik *et al.* [16] developed an attention embedded deep residual network model to identify the type of disease infected in the tomato leaves. The PlantVillage dataset is used to conduct the experiment which comprises of three different types of disease such as early blight, late blight and leaf mold. The proposed model utilises the features learned by the CNNat different processing levels. The experiment results show that the proposed model achieved a detection accuracy of 98% on the validation dataset. Jiang *et al*. [17] introduced an improved ResNet50 model to classify three different types of tomato leaf diseases such as spot blight, yellow leaf curl, and late blight. The ResNet50 model uses a leaky ReLU activation function, and to enhance the accuracy of the filter size which is modified to 11 x 11 in each convolution layer. The result showed that ResNet50 obtained an accuracy of 98% on the test dataset.

2. Proposed Method

This task describes plant classification using the newly developed compact CNN model and AlexNet using transfer learning. Thirty-two classes in the Flavia dataset are classified. Figure 3.1 shows the plant classification and validation workflow. The Flavia datasets are augmented, and the image is resized to the required size. The input image size for the proposed model is 256 X 256 X 3, and AlexNet's input image size is 227 X 227 X 3. The dataset is further divided into 80% - 20% training and test datasets. The proposed model and AlexNet are trained in a training dataset for plant species classification. The trained model is validated against test data to predict new data classes.

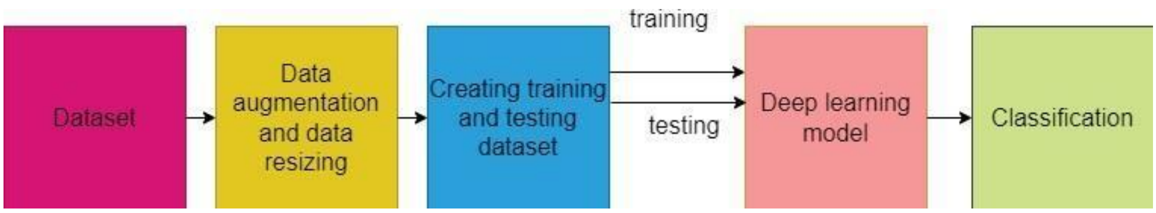


Figure 1: Workflow for classification and validation of plant leaves.

3.1 Data Pre-processing and Data Augmentation

It is necessary to follow the main steps common in the analysis, one of which is pre-processing, for the smooth operation of any algorithm and maintaining consistency in the analysis. The dataset chosen for classification is small for training a deep learning model. The total images are 600 as there are nine classes, i.e., one healthy and eight diseased. When the data is augmented, the dataset size is increased multiple times, which helps train the deep network model. In this work, different combinations of the augmented dataset are used with the rotation of 90°, 180°, and 270°, along with horizontal and vertical flips. Now, after augmentation with the said combination, the size of the dataset is 9000 images. The images in the dataset must be resized to fit the deep learning model that will be used. The required input image size for AlexNet and SqueezeNet is 227 X 227 X 3, whereas the required input image size for VGG16, GoogLeNet, MobileNet, ResNet-18, ResNet-50, and ResNet-101 is 224 X 224 X 3. The input size of the images must be satisfied for the network to fit the model.

3. Deep learning Models

Deep learning models are more intricate types of primary neural networks. The number of hidden layers in the deep learning model was enhanced compared to typical neural networks. The convolution, max-pooling, ReLU, and classification layers are the four layers that make up a CNN. The combination of these layers determines the model's design. Deep learning aids in extracting needed features from the input image. It has a high level of accuracy and can solve complicated issues quickly. The model's accuracy can be improved by modifying the layers and how they are combined in the model. The dimensionality of retrieved features is lowered with the help of a pooling layer. The fully connected layer is a dense network wherein every node is connected to every other node. It is connected before the classification layer and divides the input image into pre-determined groups or classes to predict the output. Due to their promising results, deep learning networks have been widely used in numerous areas. The ResNet model has fewer filters than VGG nets and less complexity. ResNet18 model has 11M parameters compared to VGG16 having 138M parameters. In the proposed work, the classification of TPL for nine classes is done. Transfer learning is implemented by replacing the last three layers of the pre-trained models with a fully connected layer, the softmax classifier layer. This work implements all the models using a deep learning toolbox in MATLAB 2019b.

4.1 AlexNet

AlexNet is a pre-trained 25-layer deep network that can categorize the 1000 class. This study aims to classify the nine classes of the tomato plant dataset into two categories: health and disease. Transfer learning takes place in the last three layers of the network. A flow chart of the AlexNet model proposed for classifying and predicting tomato plant diseases is shown in the Figure 2. AlexNet requires an image of this size, so the input image dataset will be expanded and resizedto 227 x 227 x 3. The AlexNet network model is trained on four combinations of training datasets, and the model is tested on the dataset. Transfer learning is essential in this model as here; there are nine output classes for classification. The last three layers are replaced by a fully connected layer that determines the desired numberof outputs to classify, followed by a softmax activation function layer and finally a classification output layer for transfer learning. The testing data is classified withthe trained model and performance parameters are evaluated.

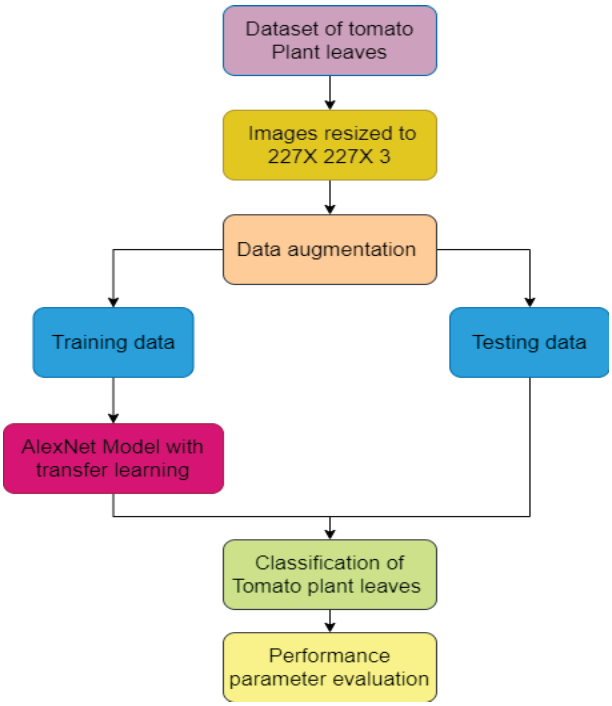


Figure 2: Flow chart for classification of Tomato plant leaves disease using AlexNet.

4.2 VGG16

VGG16 is a more complex network than AlexNet, which has more hidden layers. The network has 41 layers. Figure 3 depicts the flowchart for the proposed VGG16 model for tomato plant disease classification and prediction. This model's input image requirement is 224 X 224 X 3. As a result, after augmenting, the image input data is resized to match this size to fit the model with the required format of input image size. For VGG16, transfer learning is also performed for the nine classes of tomato plant leaves. The VGG16 model required more time to train with the dataset, which increases as the training data size increases. The trained model classifies the testing data, and performance parameters are evaluated.

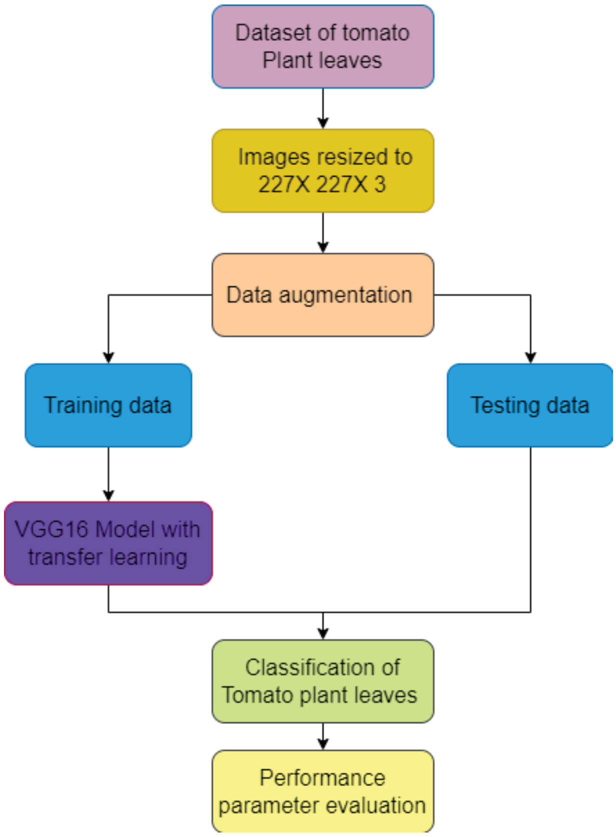


Figure 3: Flow chart for classification of Tomato plant leaves disease using VGG16.

4.3 GoogLeNet

The first convolutional network developed by was LeNet-5, a seven-level convolutional network. The name GoogLeNet was given to the winning network of the ILSVRC 2014, named after LeNet. This is the Inception 1 network, which has 144 layers. Figure 4 depicts the flowchart for the proposed GoogLeNet model for tomato plant disease classification and prediction.The GoogLeNet input size requirement is 224 X 224 X 3. The input data is sup- plemented and resized to fit this format before being used to train the model. The steps in this network's transfer learning differ from those in AlexNet and VGG16. The network was designed with computational capability, allowing it to run on a limited number of separate devices. The trained model classifies the testing data, and performance parameters are evaluated.

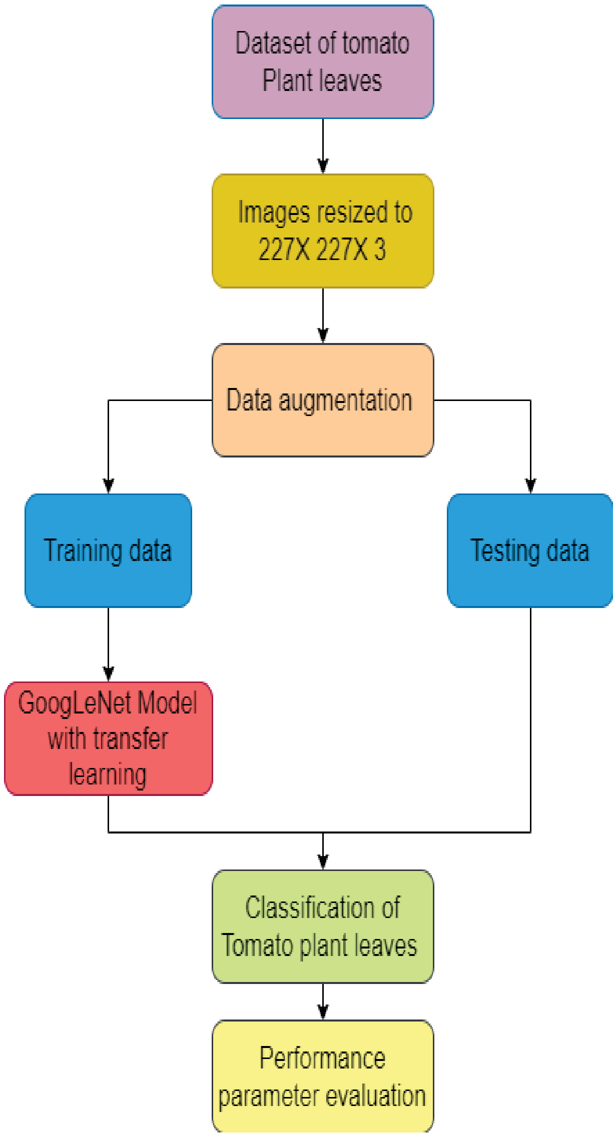


Figure 4: Flow chart for classification of Tomato plant leaves disease using GoogLeNet.

4.4 SqueezeNet

Researchers created SqueezeNet in 2016 to create a small network that can fit in computer memory. The network is a smaller 68-layer deep learning model that can be trained in less time than the other networks used in this work. Figure 5 depicts the flowchart for the proposed SqueezeNet model for tomato plant disease classification and prediction. SqueezeNet's input size requirement isthe same as AlexNet's 227 X 227 X 3. The data is supplemented and resized to the specified format before being used to train the model. The trained model classifies the testing data, and performance parameters are evaluated.

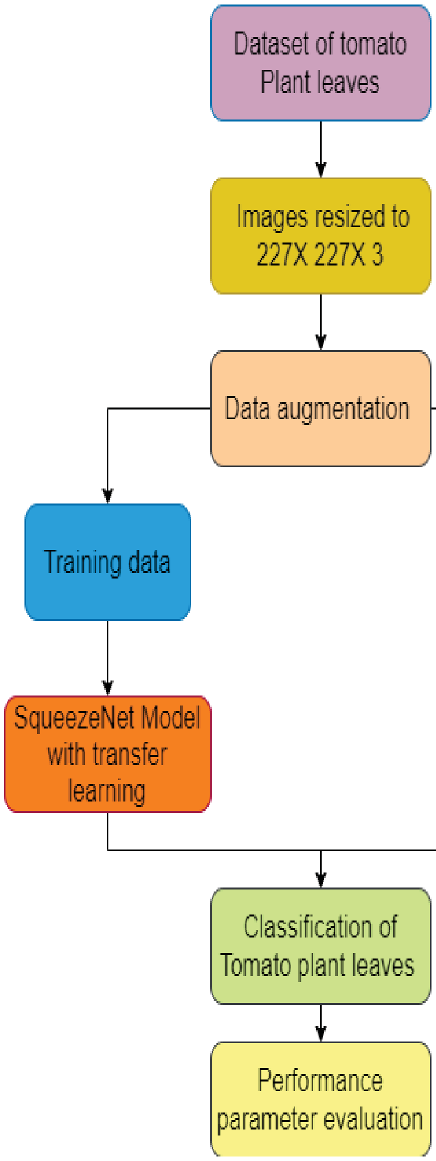


Figure 5: Flow chart for classification of Tomato plant leaves disease using SqueezeNet.

4.5 MobileNetv2

A Google team of researchers created MobileNetv2. This MobileNet model is suitable for mobile and embedded vision applications. This model has separable depth-wise convolutions layers, which helps to reduce computation time by eight to nine times. The smaller and faster MobileNet uses a width multiplier and resolution, so training the model takes less time than the GoogLeNet despite the 154-layer network. Figure 6 depicts the flowchart for the proposed MobileNetv2 model for tomato plant disease classification and prediction. The model's input size requirement is 224 X 224 X 3. The data is supplemented and resized to the specified format before being used to train the model. The trained model classifies the testing data, and performance parameters are evaluated.

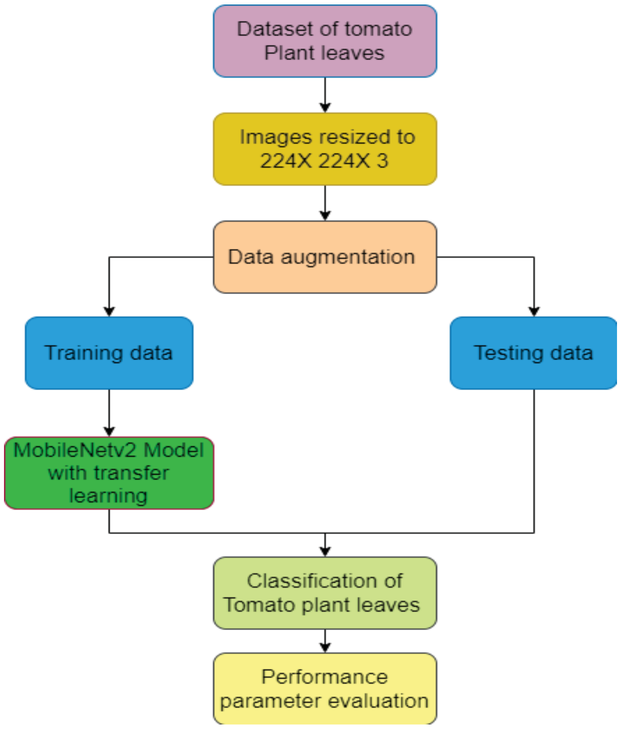


Figure 6: Flow chart for classification of Tomato plant leaves disease using MobileNetv2.

4.6 ResNet18, ResNet50, ResNet101

ResNet models can achieve impressive results by allowing for the training of hundreds or even thousands of layers. This representational ability is essential in the performance of many computer vision applications. The accuracy and ability to solve complex tasks are greatly improved when using ResNet. The ResNet mod- els attempts to address the issues of deep CNN training, saturation, and accuracy degradation. Table 1 summarises the architectures of ResNet18, ResNet50, and ResNet101.

The proposed ResNet18 architectural model with transfer learning is depicted in Figure 7. The term ResNet18 refers to a network with 18 layers of the residual network. The first layer contains 64 filters with 7 X 7 kernels. A 3 X 3 size layer and stride two are then used to max-pool. Then there's a layer group made up of four similar blocks. The first group comprises 64 3 X 3 filters, the second group is made up of 128 3 X 3 filters, the third group is made up of 256 3 X 3 filters, and the fourth group is made up of 512 3 X 3 filters. The curved lines represent the identity block, connecting two layers of varying sizes. The dotted shortcuts expand the dimensions. Finally, they are linked to nine fully connected layers for classification purposes. The descriptions for the ResNet18, ResNet50 and ResNet101 models are similar to those in Table 1. The ResNet18, ResNet50, and ResNet101 are data networks pre-trained to classify 1000 different classes. Transfer learning is used in the network's final three layers to classify the nine classes in the Tomato plant datasets into healthy and disease classes. The input image dataset is augmented and resized to the 224 X 224 X 3 size required by all the ResNet models. All of the networks are trained on four different training datasets before being tested on the testing dataset.

The models are trained using the data and the augmented datasets for the above-mentioned training- testing dataset combinations. Because there are nine output classes to classify in this model, transfer learning is critical. With transfer learning, the model is modified by replacing the last three layers with three layers indicating the number of desired classified outputs, the softmax layer, and the final output layer. Before training the models with datasets and augmented datasets, the 1000 fully connected layer is replaced by nine fully connected layers. Figure 8 depicts the flowchart for the proposed ResNet model for tomato plant disease classification and prediction. The model's input size requirement is 224 X 224 X 3. The data is supplemented and resized to the specified format before being used to train the model. The trained model classifies the testing data, and performance parameters are evaluated.

Figure 7: The Architecture model of proposed ResNet18 model with transferlearning.

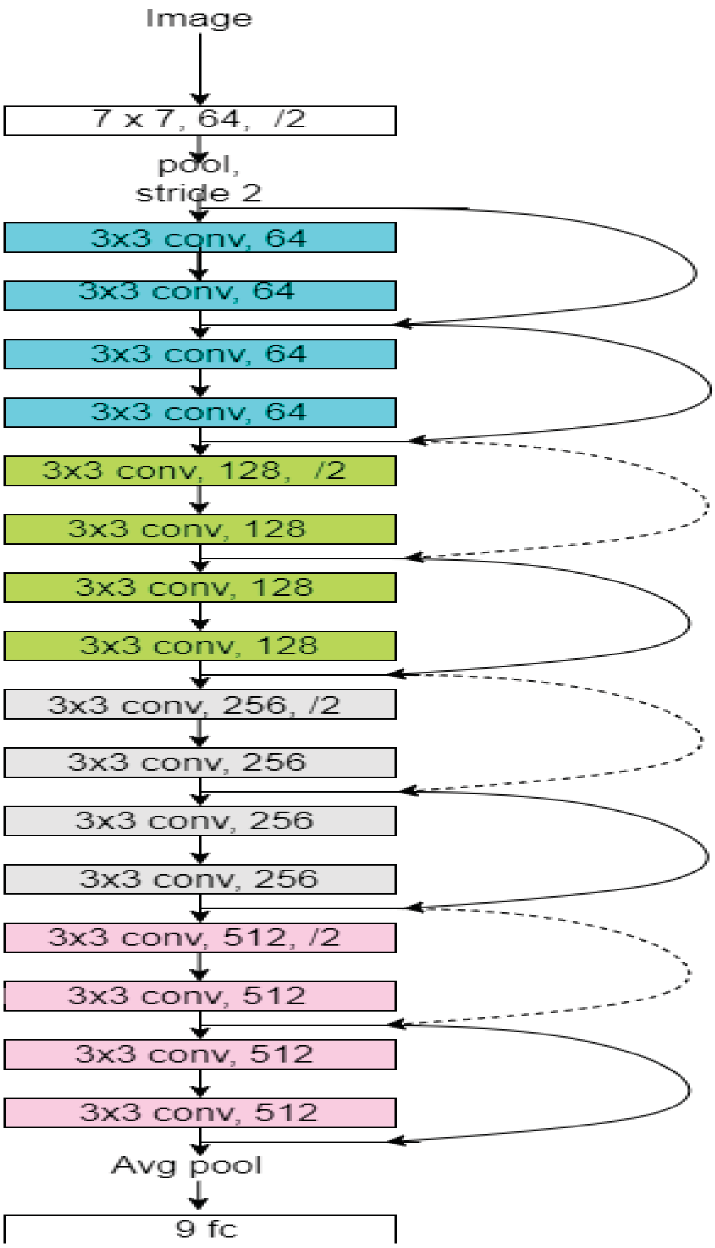


Table 1 The architecture of ResNet Models [He et al., 2016].

Layer name	Output size	18 layer	50 layer	101 layer
Conv 1	112x112	7 x 7, 64, stride 2		
Conv 2	56 x 56	3 x 3 max pool, stride 2		
		$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
Conv 3	28 X 28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$
Conv 4	14 X 14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$
Conv 5	7 X7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1 X 1	Average pool, 9 fc, softmax		

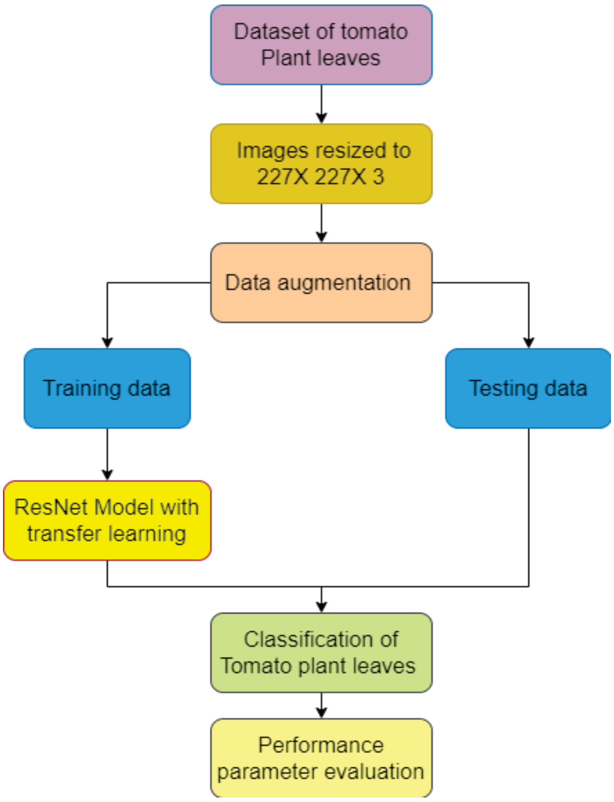


Figure 8: Flow chart for classification of Tomato plant leaves disease using ResNet odel.

4. Results and Discussion

TPL with a healthy and diseased class that is used in training the model is shown in Figure 9. The training and testing datasets are made up of both healthy and diseased plant leaves.

H BS EB LB LM

MV SLS TS YLCV

Figure 9: Tomato plant leaves images from the training dataset of PV dataset

Figure 10 shows some of the augmented images. The augmented images are divided into 80% of training data and 20% of testing data. The training and testing data consists of the images of all the tomato leaf classes.

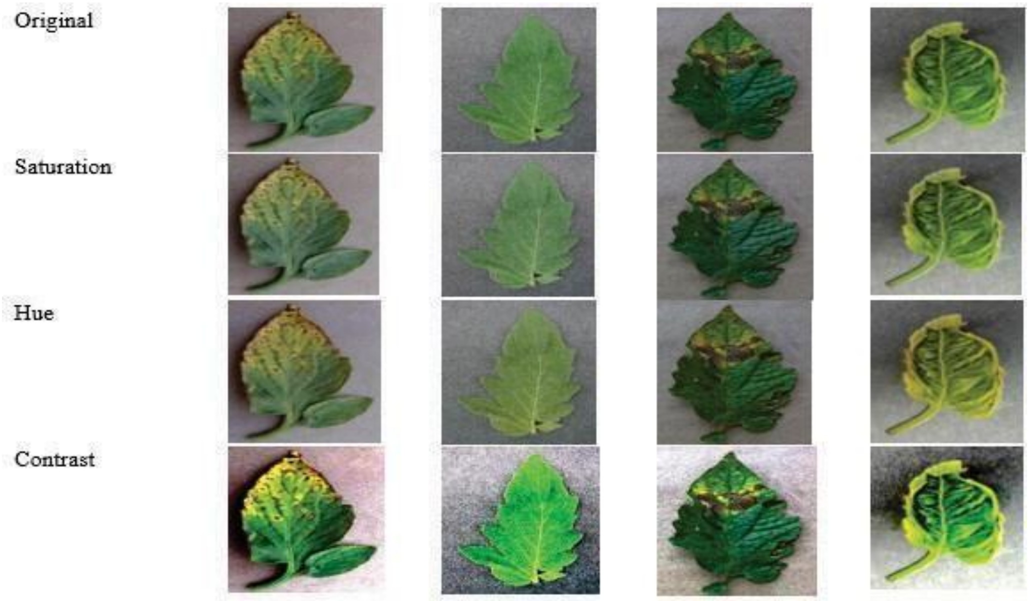


Figure 10. Pre-processed samples of a dataset of TPL.

The performance of the models was seen to be improved with the augmented dataset. The classification accuracy of the models in the classification of tomato plant leaves disease is shown in Figure 4.11. The performance of AlexNet is 99.48%, VGG16 is 99.72%, GoogLeNet is 99.23%, MobileNetv2 is 98.94%. The lowest performanceof the model in the classification of tomato plant leaves is by SqueezeNet. Theaccuracy of ResNet18 is 99.90%, ResNet101 is 99.96% and ResNet101 is 99.97%.

Table 2: Performance of proposed work in the classification of tomato plant disease

Model	Datasize	Accuracy	Model Size
AlexNet	94500	99.48%	202MB
VGG16	94500	99.72%	476MB
GoogLeNet	94500	99.23%	21.3MB
MobileNetv2	94500	98.94%	8.22MB
SqueezeNet	94500	97.96%	2.62MB
ResNet18	94500	99.90%	39.7MB
ResNet50	94500	99.96%	83.9MB
ResNet101	94500	99.97%	151MB

The classification performance of models trained with a dataset and augmented dataset is evaluated on the PV dataset using the confusion matrix, as shown in Table 2 for the AlexNet, VGG16, GoogLeNet, MobileNetv2, SqueezeNet, ResNet18, ResNet50, and ResNet101. The confusion matrix shows the information about classification and misclassification by the model. The diagonal elements show the correct classification, and the non- diagonal elements show the misclassification information. Table 3 shows the confusion matrix for the proposed model trained for 80% data set and tested for 20% of a dataset remaining. The diagonal elements marked in yellow represent the correct classification of each class.

Table 3: Confusion matrix for proposed model for PV dataset.

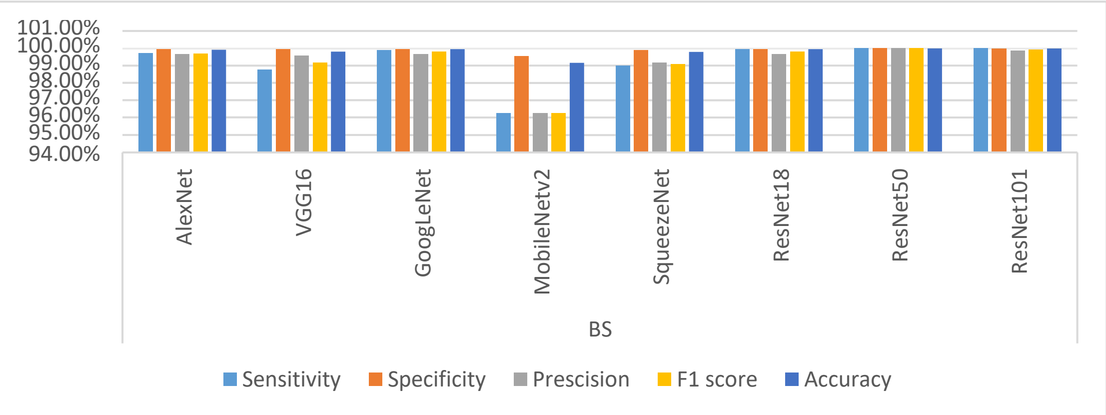
Class	BS	EB	H	LB	LM	MV	SLS	TS	YLCV
BS	238	1	0	0	0	0	0	0	1
EB	14	179	0	20	2	0	3	16	6
H	0	0	229	5	0	1	0	5	0
LB	7	5	0	223	0	0	0	2	3
LM	4	2	1	8	195	5	1	11	13
MV	0	0	0	0	0	237	1	2	0
SLS	9	6	0	18	7	7	160	22	11

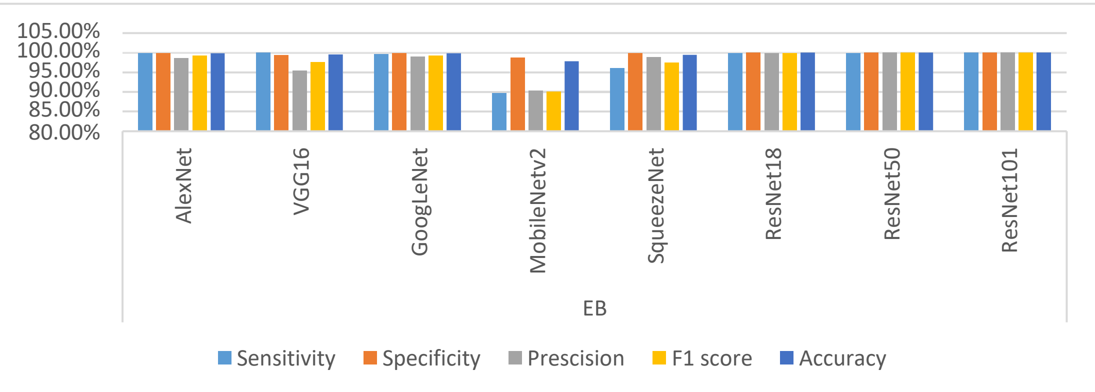
TS	1	0	0	0	0	0	0	238	1
YLCV	3	0	0	0	0	0	0	2	235

Figure 11, 12 and 13 shows the performance parameters Sensitivity, Specificity, Precision, F1 score, and Accuracy for the PV dataset for the nine tomato plant classes. Performance parameters of models for nine classes of tomato plant of PV database

5. Summary

This works aims to classify tomato plant leaves disease. The dataset used in this work is the PV dataset which is readily available. The work discusses the classification of tomato plant leaf disease belonging to eight disease classes in Indian states. The two data augmentation techniques are applied to the dataset for further use intraining the models for the classification discussed. The tomato plant dataset withhealthy and eight disease classes BS, EB, LB, LM, MV, TS, SLS, and YLCV are classified with AlexNet, VGG16, GoogLeNet, MobileNetV2, SqueezeNet, ResNet18, ResNet50, and ResNet101. The effect of data augmentation on models' performance is discussed here. The proposed model achieved high accuracy with small model size and a short training time.





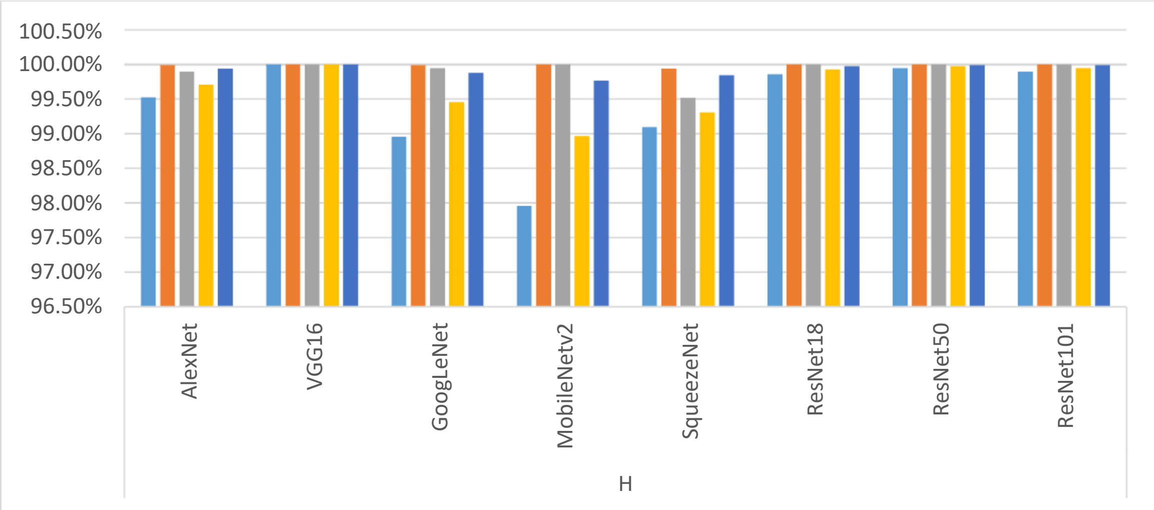
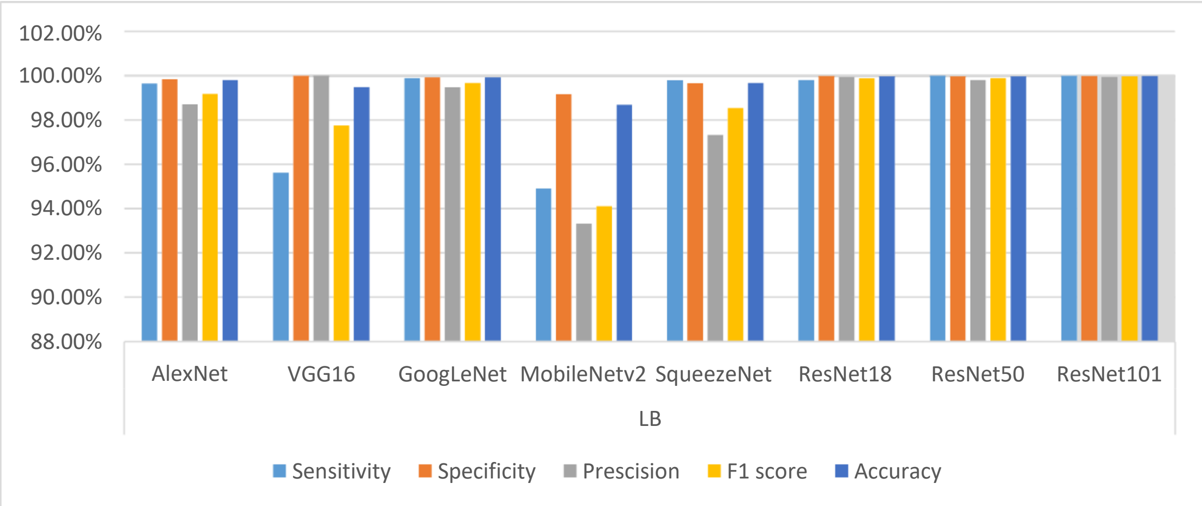
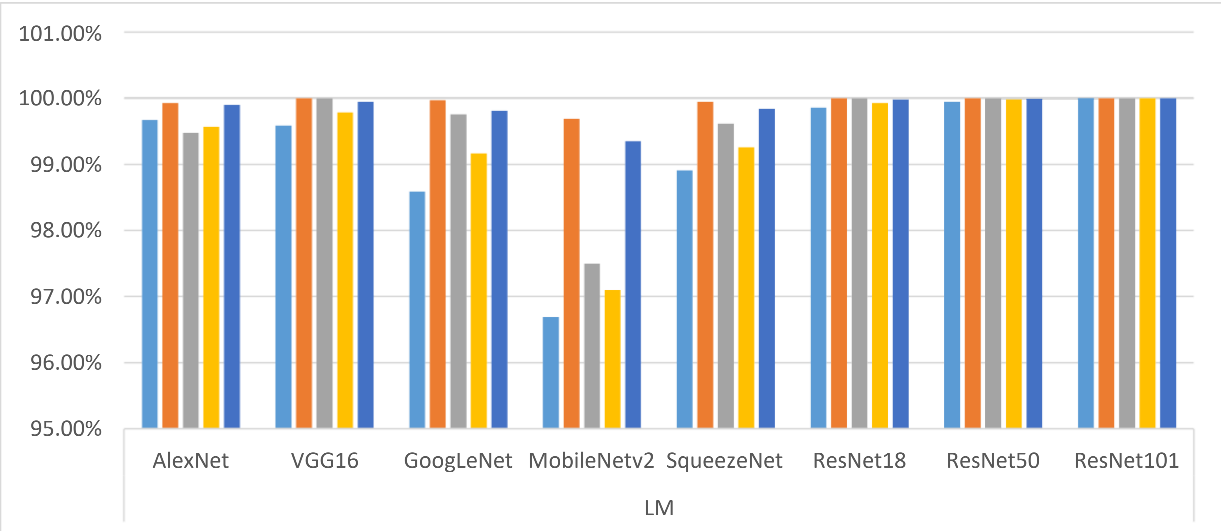


Figure 11: shows the performance parameters Sensitivity, Specificity, Precision, F1 score, and Accuracy for the PV dataset for the nine tomato plant classes





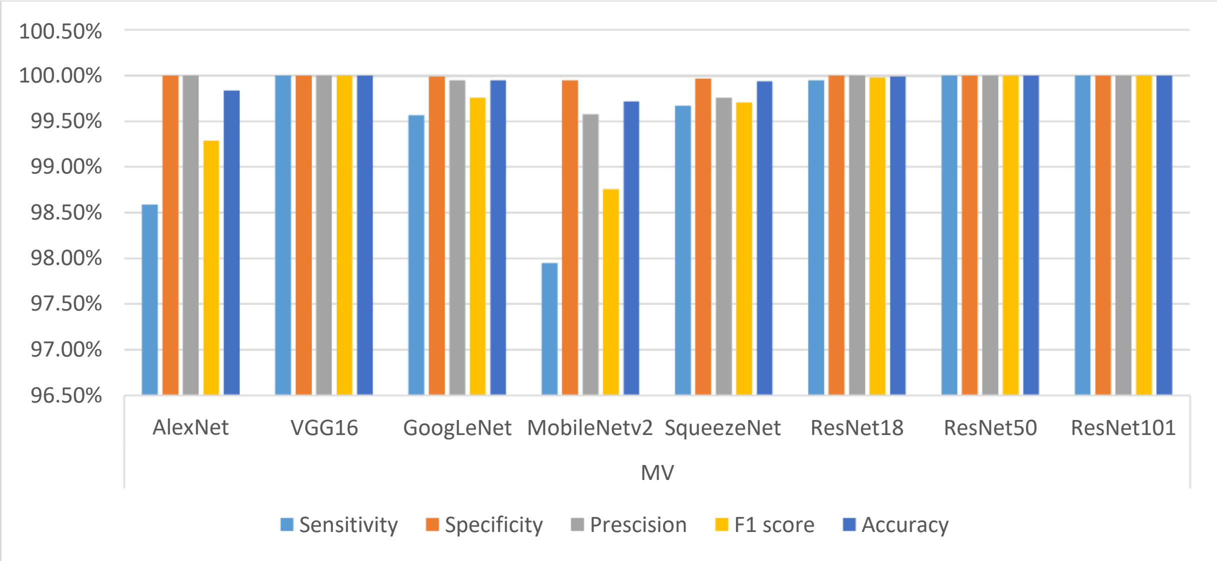
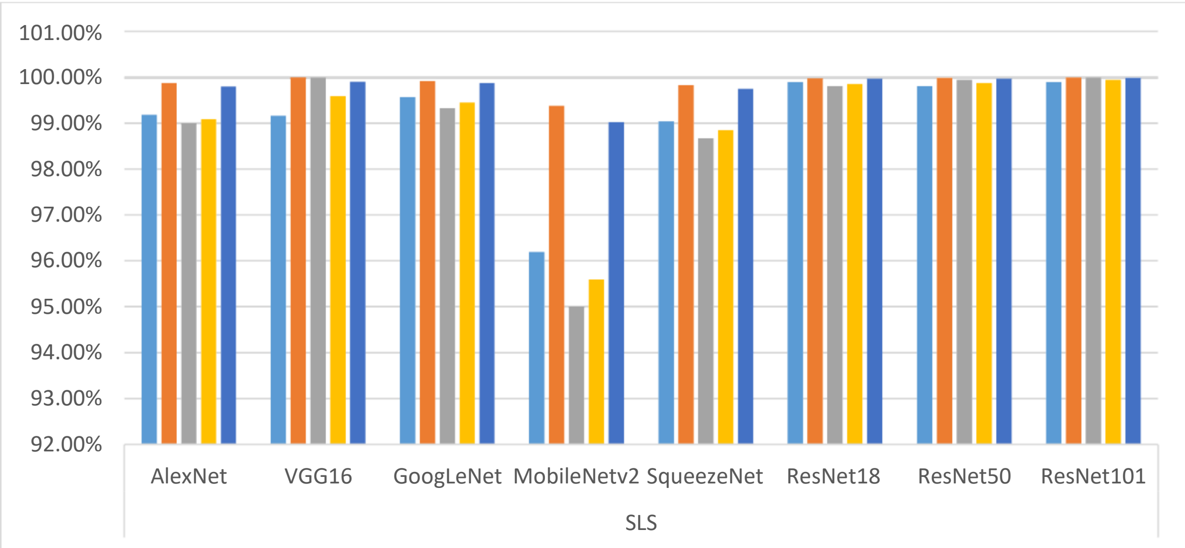
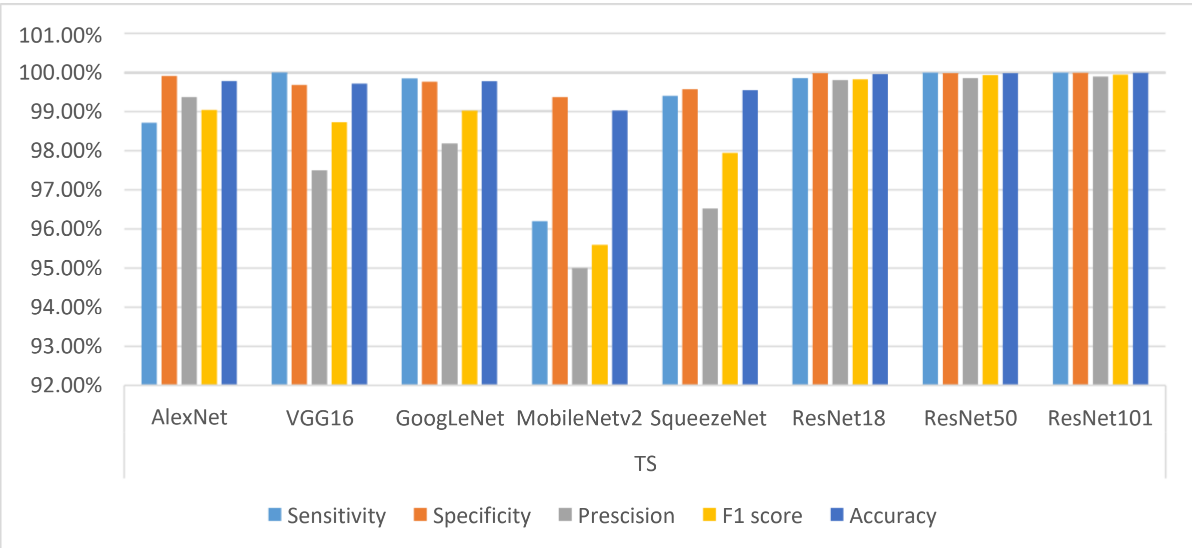


Figure 12: shows the performance parameters Sensitivity, Specificity, Precision, F1 score, and Accuracy for the PV dataset for the nine tomato plant classes



\



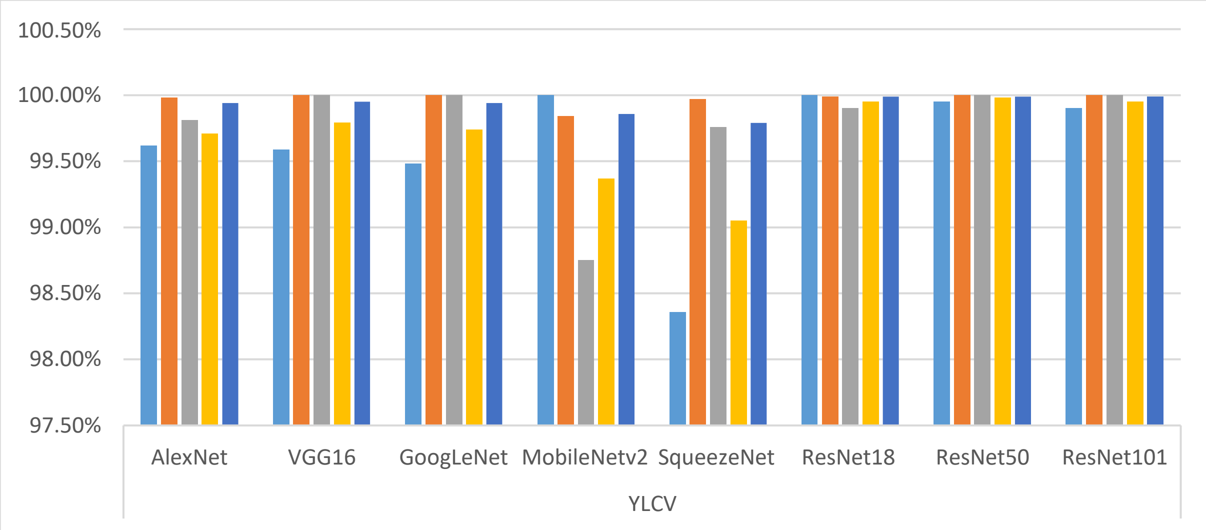


Figure 13: shows the performance parameters Sensitivity, Specificity, Precision, F1 score, and Accuracy for the PV dataset for the nine tomato plant classes

References

1. Ma, J, Li, X , Wen, H, Chen, Y , Fu, Z & Zhang, L 2015, 'Monitoring video capture system for identification of greenhouse vegetable diseases', Transactions of the Chinese of the Society for Agricultural Machinery, vol. 46, no. 3, pp. 282-287.
2. Barbedo, JGA 2013, 'Digital image processing techniques for detecting, quantifying and classifying plant diseases', SpringerPlus, vol. 2, no. 1, pp. 1-12.
3. Bengio, Y 2013, 'Deep learning of representations: Looking forward', In: International Conference on Statistical Language and Speech Processing. Springer, pp. 1–37.
4. Basavaiah, J & A nthony, AA 2020, 'Tomato Leaf Disease Classification using Multiple Feature Extraction Techniques', Wireless Personal Communications, vol. 115, no. 1, pp. 633-651.
5. Annabel, LSP & Muthulakshmi, V 2019, 'AI-Powered Image-Based Tomato Leaf Disease Detection', In 2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), pp. 506-511.
6. Foysal, MFA, Islam, MS, Abujar, S & Hossain, SA 2020, 'A novel approach for tomato diseases classification based on deep convolutional neural networks', In Proceedings of International Joint Conference on Computational Intelligence, Springer, Singapore, pp. 583-591.
7. Ruedeeniraman, N, Ikeda, M & Barolli, L 2020, 'Performance evaluation of V egeCare tool for potato disease classification', In International Conference on Network-Based Information Systems, Springer, Cham, pp. 470-478.
8. Sardogan, M, Tuncer, A & Ozen, Y 2018, 'Plant leaf disease detection and classification based on CNN with LV Q algorithm', In 2018 3rd International Conference on Computer Science and Engineering (UBMK), pp. 382-385.
9. Rubanga, DP, Loyani, LK , Richard, M & Shimada, S 2020, 'A Deep Learning Approach for Determining Effects of TutaA bsoluta in Tomato Plants', arX iv preprint arX iv:2004.04023.

10. Chen, X , Zhou, G, Chen, A, Y i, J, Zhang, W & Hu, Y 2020, 'Identification of tomato leaf diseases based on combination of ABCK -BWTR and B-ARNet', Computers and Electronics in Agriculture, vol. 178, pp. 105730.
11. Agarwal, M, Singh, A, Arjaria, S, Sinha, A & Gupta, S 2020, 'ToLeD: Tomato Leaf Disease Detection using Convolution Neural Network', Procedia Computer Science, vol. 167, pp. 293-301.
12. Liu, J & Wang, X 2020, 'Tomato diseases and pests detection based on improved Y olo V 3 convolutional neural network', Frontiers in plant science, vol. 11, pp. 898
13. Liu, J & Wang, X 2020, 'Early recognition of tomato gray leaf spot disease based on MobileNetv2-Y OLOv3 model', Plant Methods, vol. 16, pp. 1-16.
14. Gadekallu, TR, Rajput, DS, Reddy, MPK , Lakshmanna, K , Bhattacharya, S, Singh, S & Alazab, M 2020, 'A novel PCA–whale optimization-based deep neural network model for classification of tomato plant diseases using GPU', Journal of Real-Time Image Processing, pp. 1-14.
15. Prabhakar, M, Purushothaman, R & Awasthi, DP 2020, 'Deep learning based assessment of disease severity for early blight in tomato crop', Multimedia Tools and Applications, vol. 79, no. 39, pp. 28773-28784.
16. K arthik, R, Hariharan, M, Anand, S, Mathikshara, P, Johnson, A & Menaka, R 2020, 'A ttention embedded residual CNN for disease detection in tomato leaves', Applied Soft Computing, vol. 86, pp. 105933
17. Jiang, D, Li, F, Y ang, Y & Y u, S 2020, 'A Tomato Leaf Diseases Classification Method Based on Deep Learning', In 2020 Chinese Control A nd Decision Conference (CCDC), pp. 1446-1450.