Recognize and classify illnesses on tomato leaves using EfficientNet's Transfer Learning Approach with different size dataset

Pratik Buchke
pratik.buchke@gmail.com

Vellore Institute of Technology University
V R Mayuri
Vellore Institute of Technology University

Research Article

Keywords: Image Processing, CNN, Transfer Learning, EfficientNet, Classification, Plant Disease

Posted Date: July 14th, 2023

DOI: https://doi.org/10.21203/rs.3.rs-3149045/v1

Additional Declarations: No competing interests reported.

Version of Record: A version of this preprint was published at Signal, Image and Video Processing on April 26th, 2024. See the published version at https://doi.org/10.1007/s11760-024-03188-z.

Abstract

This study focuses on the remarkable progress made by the agricultural sector in utilizing image processing techniques for early detection and classification of leaf plant diseases. Timely identification of diseases is crucial, but it often poses a challenge for the human eye to discern subtle differences. To address this issue, the researchers propose a novel approach that employs EfficientNet, a deep learning model, to accurately recognize various diseases affecting tomato plant leaves. Transfer learning is applied to three different datasets comprising 3000, 8000, and 10,000 images of diseased tomato leaves. The experimental results demonstrate impressive overall accuracies of 97.3%, 99.2%, and 99.5% when using 3000, 8000, and 10,000 images, respectively, for the detection of common tomato plant diseases. This research underscores the effectiveness of image processing and deep learning techniques in achieving precise and efficient detection of tomato leaf diseases. It significantly contributes to the advancement of precision agriculture and enhanced crop management practices.

I. Introduction

India's economy depends heavily on the agricultural sector, which generates a sizeable share of both employment and GDP. Plant diseases, on the other hand, present a significant problem by impeding organic development and harming leaves, stems, and seeds. It's essential to find plant diseases early on to boost overall productivity. Specialist manual diagnosis of leaf diseases is antiquated, ineffective, and time-consuming. In order to increase agricultural output, effective methods for plant disease identification, particularly through leaf inspection, are crucial.

In order to identify and classify plant leaf diseases, this research provides a machine learning-based strategy using proposed model. The model facilitates the diagnosis and classification of plant illnesses by fusing machine learning with digital image processing techniques. By avoiding the problems associated with too large, deep models or high resolutions that may result in inefficiency and parameter saturation, EfficientNet offers a more methodical scaling strategy.[9, 12, 15]

The performance evaluation on the testing set utilizing the Plant Village dataset with growing dataset size and the development of a more accurate disease recognition model for tomato plant leaves are the main contributions of this study. This research proposes a novel method that makes use of a public dataset of 10,000 images of healthy and diseased tomato leaves, and with different size in contrast to earlier methods that rely on pre-trained models. On a different test set, the model correctly identified nine tomato leaf diseases with a 99.5% accuracy rate. The usefulness and practicality of the suggested approach in detecting illness on tomato plant leaves are strongly supported by these data.

An EfficientNet with changes in training layer is used to categorize diseases into one of the classifications. Using a dataset of tomato leaves, we put our suggested model to the test. It is structured as follows: The Literature Review is included in Section II. Section III looked at the proposed system for recognizing and classifying leaf diseases. Laboratory conditions are covered in Section IV. Results and discussion are included in Section V. The conclusion is covered in Section VI.

II. Literature Review

Numerous studies that have been conducted on identifying leaf diseases were covered in this section. The development of computer-aided leaf disease detection in a range of plants was the focus of this field's study. Computer vision and machine learning have been widely used by researchers to identify plant leaf disease in earlier years.

Anil A. Bharate[1].In this article, they assess methods developed by several image processing researchers for the purpose of identifying plant diseases. Research on spotting plant diseases early in crops like tomato, apple, grapes, pepper, and pomegranate are covered in this article.

Jayme Garcia Arnal Barbedo[2]This essay analyses each of those difficulties, focusing on the issues they could raise along with how they might have impacted earlier suggested solutions. There are a few suggested potential solutions that might perhaps overcome at least some of those difficulties, but only under certain circumstances.

J.Nithiswara Reddy[3]With minimal computing effort, the proposed method can greatly support a precise diagnosis of leaf diseases. They developed framework software in Matlab to identify plant leaf diseases by using methods for processing images. The program is intended to enable even a person without prior knowledge of plants or their diseases to spot diseased leaves. By applying k-means clustering, the afflicted portion of the plant leaf was located. Obtaining pictures, processing images before segmentation and feature extraction, and SVM classification are all included in the diseased recognition section.

Durjoy Sen Maitra[4]This paper aims to demonstrate a feature extraction approach that may accustomed to any character recognition problem. Here, we've demonstrated that a from any other character set, CNN the ability to extract characteristics after being trained on a sufficiently big class issue, and the resulting system is still capable of delivering high recognition accuracies.

Nikhil Shah[5]The main goal of the study is to identify the various diseases that affect cotton using an artificial neural network tool, which applies an image pre-processing approach to pictures. Based on color changes on the image, the main area of the affected leaf is highlighted, and the disease's type is determined using data.

T. Rumpf [6] This work's main contribution a process for early diagnosisand isolation of sugar beet illnesses based on spectral vegetation indicators and Support Vector Machines. The objective of the research was to detect infections in sugar beet leaves before visible symptoms appeared.

MiaomiaoJi[7],Thisresearch suggests a hybrid PSO-based ANN model (PSO-ANN) for the problem of soybean diseases identification based on the condition of the environment and different features of the soybean, such as the plant stand, the leaves and the seed, etc.

Muzaiyanah Binti Ahmad Supian [8],For the benefit of agriculturalists working in the agrarian industry, this study investigates image processing methods a means to locate and categorization of leaf plant diseases.There are several phases included in it, including acquisition of images, image processing, extraction of features from segments, followed by categorization.

Jayme G.A. Barbedo[9],The main elements that influence this article which examines the construction and performance of deep neural networks used for plant pathology. Realistic conclusions on the topic should result from an extensive analysis of the issue that highlights its benefits and drawbacks.

Bin Liu[10],This study demonstrates how the picture-generating technique suggested the ability to strengthen the convolutional neural network model and how the suggested deep learning model provides a better option for Disease prevention for apple leaf diseases with more precision and speedier convergence.

Table 1
Comparison of research papers

Method	Performance	Limitation
[1] ANN along with SVM was used for crop disease classification (2017)	NA	Review Paper
[2]Proposed solution with busy background and different scenarios (2016)	NA	All proposed methods are expected to operate under certain time constraints.
[3] K-means clustering with SVM & KNN classifier (2019)	Accuracy 94% (SVM) 85% (KNN)	Better segmentation technique require, working on small dataset
[4] CNN with SVM classifier (2015)	Accuracy 95%	This approach is slow and time-consuming
[5]ANN with t-test (2019)	NA	Performance needs further improvement
[6] SVM classifier with Vis & PCs (2010)	Accuracy 97% & 86%	With multiple classifications accuracy is less.
[7] ANN for classification & PSO for identification (2019)	Accuracy 92%	Performance can be improved
[8]Review Paper (2019)	NA	Various frameworks wrongly recognize and classify plant diseases.
[9] A CNN-based architecture was presented to localize and categorize the tomato crop Disease (2018)	Accuracy 87%	This framework is not robust for noisy images
[10] Deep CNN with Google Net approach (2018)	Accuracy 97.62%	Working with 4 type of diseases
[11] A deep Siamese network together with KNN classifier was used for plant disease classification (2020)	Accuracy 96%	This method is suffering from the problem of over-fitting for a large-size dataset
[12] The DL framework namely AlexNet along with the KNN classifier was used to classify the tomato leaves as being healthy or affected (2020)	Accuracy 76.1%	This approach is slow and time-consuming
[13] The HOG approach with the RF classifier was employed to categorize the diseased plant samples into various classes (2018)	Accuracy 70.14%	Performance needs further improvement
[14] The K-means clustering, GLCM methods along with SVM classifier were utilized to classify turmeric leaf diseases (2019)	Accuracy 91%	Classification performance degrades for samples having huge brightness variations

III. Proposed Methodology for Plant Leaf Disease Detection and Classification

Convolution neural networks can be built up to increase accuracy by adding more layers, and their resource costs are fixed. The standard approaches to model scaling, however, are inconsistent. Some models scale in depth, while others scale in width. Some models merely consume higher-resolution images to obtain better results. When models are scaled arbitrarily, it often results in little or no performance improvement and requires extensive human tweaking. EfficientNet uses a technique known as compound coefficient to quickly and simply scale up models. Instead of arbitrarily growing width, depth, or resolution, compound scaling consistently scales each dimension with a preset fixed set of scaling factors. By combining scaling with AutoML, the developers of EfficientNet created seven models in various dimensions that outperformed state-of-the-art convolution neural networks in terms of accuracy and efficiency.

Model Scaling

According to the logic, scaling all three dimensions—width, depth, and picture resolution—while taking into account the various resources available, can best increase the model's performance overall. Scaling one dimension can help improve model performance. The compound scaling method is shown in figure.

1. Scaling Convnet- It can be described as modifying the network's dimensions to improve performance based on the most popular definitions. Depth, width, and resolution make up the dimensions.
2. Compound scaling- The authors of EfficientNet suggest starting with a baseline network (N) and concentrating on expanding its length (L), width (C), and resolution (W, H) while maintaining the baseline design. This differs from the typical method of looking for the ideal layer architecture. Thus, choosing the ideal width (w), depth (d), and resolution (r) coefficients within the constraints of the resources available to maximize the accuracy of the network (memory and number of feasible operations (FLOPS)) is the definition of the optimization issue.

$$\max_{d,w,r} \quad Accuracy(\mathcal{N}(d,w,r))$$
$$\text{Memory}(\mathcal{N}) \leq \text{target\_memory}$$
$$\text{FLOPS}(\mathcal{N}) \leq \text{target\_flops} \qquad (1)$$

In order to further reduce the search space < L,C,W,H>, the authors also suggested to restrict that all layers must be scaled uniformly using a constant ratio. Thus, the dimensions of the network are defined as:

$$\textbf{depth: } d = \alpha^{\phi}$$
$$\textbf{width: } w = \beta^{\phi}$$
$$\textbf{resolution: } r = \gamma^{\phi}$$

The compound coefficient Φ, controlled by the user, determines the number of available resources. α, β, and γ are constants found through grid search, which allocate these resources to the network's depth, width, and resolution respectively.

It is also important to mention that the authors noticed that the FLOPS of a regular convolution operation are proportional to d, w^2, r^2. Since convolution operations dominate the computation cost in ConvNets, using compound scaling on a Convnet increases the number of FLOPS by $(\alpha.\beta^2.\gamma^2)\Phi$, thus the constraint $\alpha.\beta^2.\gamma^2 \approx 2$, to increase the total FLOPS by 2Φ.

3. EfficientNet architecture

Compound scaling, as previously said, enhances the network's width, depth, and resolution rather than altering the operations carried out within a layer of the network. Following is the architecture of the model-

MBConv

Skip connections are used by residual blocks to link a convolutional block's start and finish. The channels are wide at the start of the convolutional block, get smaller as the block depth rises, and then get wider again at the end due to the additional information. Wide->narrow->wide is the pattern for a typical residual block in terms of the number of channels.[18]

The pattern of an inverted residual block, however, is the opposite of that of a regular residual block; it means narrow->wide->narrow. MBConv enhances efficiency and adaptability of CNNs for mobile platforms using Depth-wise Separable Convolution. The remaining channels are compressed at the beginning and end of the block using a 1x1 convolution, followed by a 3x3 depth-wise convolution to restrict the parameters.

Squeeze and Excitation (SE) Block

SE is a CNN component that improves interdependencies between channels by dynamic feature channel-wise recalibration, giving relevant channels more weight than unimportant ones. View the illustration below.

The following structure is the result of EfficientNet applying the SE block along with the MBConv block. The initial component of each network is its stem, after which all architecture experimentation, which is common to all eight models and the top layers, starts.

Following that, each of them has seven blocks. As we progress from EfficientNetB0 to EfficientNetB7, the number of these blocks' sub-blocks increases, with a different amount being present in each block. The architecture will be built using 5 modules. These modules are then joined to create sub-blocks, which will be utilized in the blocks in a particular manner.[11, 21]

IV. Experimental Settings

The approach was used on a dataset from Plant Village that included 3000, 8000, and 10000 images of ten tomato leaf diseases. The model was created using Python's Keras neural network. Training used 2000, 7000, and 9000 images, while testing used 1000. On Google Colab, the tests were carried out using a GPU and an Intel Core i7-4010U processor.

In this study, we will use the EfficientNet on the Plant Village dataset to do multi-class image classification. To implement it as a transfer learning model, we have used the EfficientNet-B3.The Plant Village dataset is a publically available image data set. The dataset has 10,000 color images, 32x32 in size, divided into 10 classes with 900 images training & 100 images validation in each category. The 10 different classes represent Bacterial_spot, Early_blight, Late_blight, Leaf_Mold, Septoria_leaf_spot, Spider_mites, Two-spotted_spider_mite, Target_Spot, Tomato_Yellow_Leaf_Curl_Virus, Tomato_mosaic_virus, healthy. There are 9000 training images and 1000 test images in this dataset.

V. Results

Modules are imported, images are taken from the Plant Village dataset directory, and the trim function is used to balance the dataset. There are internal generators for training, testing, and validation. There are defined operations for showing samples, training models, monitoring, charting predictions, Confusion Matrix, and Classification Report. A dataframe is trimmed using the max_samples and min_samples for each class in the trim function. Classes with fewer than min_samples images are excluded. The dataset is divided into three groups (2000, 7000, and 9000 photos), and each category is trained independently.

Function that shows training images

The foundation model should initially not be trainable, according to experts. The model is then fine-tuned by making the underlying model trainable and running extra epochs after training for a certain number of them. It will converge faster and have a lower validation loss.

Function that plot the training data

The Evaluation Index& Predictions on the test set

In order to evaluate the performance, average accuracy evaluation index recognized in the field of image classification is used to evaluate the classification results, including Precision (PPV), Recall (TPR), F1 Score (F1).

$PPV = T_p / T_p + F_p$ (2)

$TPR = T_p / T_p + F_n$ (3)

F1 = 2 x (PPV x TPR / PPV + TPR) (4)

Where, the number of positive samples that actually turn out to be positive samples is known as the true positive rate (T_p), whereas the false positive rate (F_p) and false negative rate (F_n), respectively, reflect the number of negative samples that actually turn out to be negative samples.[8, 12].

A function is defined that takes a test generator and an integer test_steps and generates predictions on the test set including a confusion matrix and a classification report.[13, 18, 24]

Analysis of Model Performance

In order to reduce training time the number of samples per class was limited to 200 images then with 700 images and then finally with 900 images. We could have used the trim function with max_samples = 200 then 700 and then 900 to get different training accuracy. The image size of the original images was 600 X 600 but the model was trained with 200 X 200 images again to reduce training time. Overall the model did well with an average F1 score of 99.5%. We ran for 12 epochs and the validation loss was still decreasing with about a 8% reduction in epoch 12. So we could run more epochs and probably achieve a better F1 score.

Table 2
Comparison of Results with different size dataset

Model	Dataset Samples	Accuracy	Precision	Recall	F1-score
ANN	10000	69%	0.89	0.79	0.84
CNN	10000	94%	0.92	0.94	0.92
Proposed Model	3000	97.30%	0.9748	0.9730	0.9731
	8000	99.20%	0.9922	0.9920	0.9922
	10000	99.50%	0.9950	0.9950	0.9950

VI. Conclusion

The Indian agricultural industry heavily relies on tomato crops, making it crucial to identify and describe their diseases. This research aims to achieve this using a convolutional neural network model, EfficientNet, and the Plant Village dataset. The proposed research utilized an EfficientNet convolutional neural network model and the Plant Village dataset to identify and describe tomato leaf diseases. The model achieved impressive accuracies of 97.3%, 99.2%, and 99.5% with varying dataset sizes, showing its potential as a low-resource method for disease classification. The implementation's simplicity and smaller training images required minimal hardware and fewer parameters, yet delivered comparable results to conventional techniques. Further experiments may explore different learning rates and optimizers to enhance performance.

Declarations

Ethical Approval

This research study, involving human and/or animal subjects, was conducted in compliance with ethical principles and guidelines. The study protocol and procedures were reviewed and approved by an independent ethics committee.

Consent to Participate: All participants provided informed consent, understanding the purpose, procedures, potential risks, and benefits of their participation in the study.

Consent to Publish: Participants were also informed that the research findings may be published while ensuring their privacy and anonymity.

Competing interests

We declare that we have no competing interests associated with this research study. The authors affirm that this research was conducted in an unbiased manner and that there are no relationships or conflicts of interest that could compromise the integrity, impartiality, or validity of the findings presented in this paper. Furthermore, there are no financial or other contractual agreements that could be perceived as influencing the interpretation or reporting of the results. This statement is made to ensure transparency and to maintain the highest standards of integrity in the publication of this research study.

Authors' contributions

Conceptualization- P.B. (Pratik Buchke) and A.M. (AVR Mayuri)

Methodology- P.B. and A.M.

Software- P.B.

Validation- P.B. and A.M.

Formal analysis- P.B. and A.M.

Figures- P.B.

Investigation- P.B. and A.M.

Resources- P.B.

Data curation- P.B. and A.M.

Writing—original draft preparation- P.B.

Writing—review and editing- P.B. and A.M.

All authors reviewed the manuscript.

Funding - Not Applicable

Availability of data and materials

https://www.kaggle.com/datasets/kaustubhb999/tomatoleaf

The dataset has 10,000 colour images, divided into 10 classes with 900 images training & 100 images validation in each category.

References

1. Bharate, Anil A., and M. S. Shirdhonkar. "A review on plant disease detection using image processing." In 2017International Conference on Intelligent Sustainable Systems (ICISS), pp. 103-109. IEEE, 2017.
2. Barbedo, Jayme Garcia Arnal. "A review on the main challenges in automatic plant disease identification based onvisible range images." Biosystems engineering 144 (2016): 52-60.
3. Reddy, J. Nithiswara, Karthik Vinod, and AS RemyaAjai. "Analysis of classification algorithms for plant leaf disease detection." In 2019 IEEE international conference on electrical, computer and communication technologies(ICECCT), pp. 1-6. IEEE, 2019.
4. Maitra, Durjoy Sen, Ujjwal Bhattacharya, and Swapan K. Parui. "CNN based common approach to hand written character recognition of multiple scripts." In 2015 13th International Conference on Document Analysis andRecognition (ICDAR), pp. 1021-1025. IEEE, 2015.
5. Shah, Nikhil, and Sarika Jain. "Detection of disease in cotton leaf using artificial neural network." In 2019 AmityInternational Conference on Artificial Intelligence (AICAI), pp. 473-476. IEEE, 2019.
6. Rumpf, T., A-K. Mahlein, U. Steiner, E-C. Oerke, H-W. Dehne, and L. Plümer. "Early detection and classification ofplant diseases with support vector machines based on hyperspectral reflectance." Computers and electronics inagriculture 74, no. 1 (2010): 91-99.
7. Ji, Miaomiao, Peng Liu, and Qiufeng Wu. "Feasibility of Hybrid PSO-ANN Model for Identifying SoybeanDiseases." International Journal of Cognitive Informatics and Natural Intelligence (IJCINI) 15, no. 4 (2021): 1-16.
8. Supian, Muzaiyanah Binti Ahmad, HizmawatiMadzin, and ElmalianaAlbahari. "Plant Disease Detection and Classification Using Image Processing Techniques: a review." In 2019 2nd International Conference on AppliedEngineering (ICAE), pp. 1-4. IEEE, 2019.
9. Barbedo, Jayme GA. "Factors influencing the use of deep learning for plant disease recognition." Biosystemsengineering 172 (2018): 84-91.
10. Liu, Bin, Yun Zhang, DongJian He, and Yuxiang Li. "Identification of apple leaf diseases based on deep convolutional neural networks." Symmetry 10, no. 1 (2018): 11.
11. Argueso D et al (2020) Few-Shot Learning approach for plant disease classification using images taken in the field.Comput Electron Agric 175:105542

12. Batool A et al (2020) Classification and Identification of Tomato Leaf Disease Using Deep Neural Network. In: 2020 International Conference on Engineering and Emerging Technologies (ICEET).IEEE
13. TmP et al (2018) Tomato leaf disease detection using convolutional neural networks. In: 2018 Eleventh InternationalConference on Contemporary Computing (IC3). IEEE
14. Kuricheti G, Supriya P (2019) Computer Vision Based Turmeric Leaf Disease Detection and Classification: A Stepto Smart Agriculture.In: 2019 3rd International Conference on Trends inElectronics and Informatics (ICOEI). IEEE
15. Cuong Nguyen, Tal Hassner, Matthias Seeger, and Cedric Archambeau. Leep: A new measure to evaluatetransferability of learned representations. In Inter- national Conference on Machine Learning, pages 7294{7305.PMLR, 2020.
16. Thomas Mensink, Jasper Uijlings, Alina Kuznetsova, Michael Gygli, and Vittorio Ferrari. Factors of infuence for transfer learning across diverse appearance domains and task types. arXiv preprint arXiv:2103.13318, 2021.
17. Basil Mustafa, Aaron Loh, Jan Freyberg, Patricia MacWilliams, Megan Wilson, Scott Mayer McKinney, MarcinSieniek, Jim Winkens, Yuan Liu, Peggy Bui, et al. Supervised transfer learning at scale for medical imaging. arXiv preprint arXiv:2101.05913, 2021.
18. Kaustubh, B. Tomato Leaf Disease Detection Tomato Leaf Disease Detection using CNN. Available online: https://www.kaggle.com/kaustubhb999/tomatoleaf (accessed on 21 September 2021).
19. Gao, R.;Wang, R.; Feng, L.; Li, Q.;Wu, H. Dual-branch, efficient, channel attention-based crop diseaseidentification. Comput.Electron. Agric. 2021, 190, 106410.
20. Arya, S.; Singh, R. A Comparative Study of CNN and AlexNet for Detection of Disease in Potato and Mango leaf.In Proceedings of the 2019 International Conference on Issues and Challenges in Intelligent Computing Techniques(ICICT), Ghaziabad,India,27–28 September 2019; IEEE: Piscataway, NJ, USA, 2019.
21. Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and CLawrence Zitnick. Microsoft coco: Common objects in context. In European conference on computer vision, pages 740{755. Springer, 2014.
22. Chauhan, Sumika, Manmohan Singh, and Ashwani Kumar Aggarwal. "Data Science and Data Analytics: Artificial Intelligence and Machine Learning Integrated Based Approach." Data Science and Data Analytics: Opportunities and Challenges (2021): 1.
23. Sun, J.; Tan, W.; Mao, H.; Wu, X.; Chen, Y.; Wang, L. Recognition of multiple plant leaf diseases based on improved convolutional neural network. Trans. Chin. Soc. Agric. Eng. 2017, 33, 209–215.
24. Kaushik, M.; Prakash, P.; Ajay, R.; Veni, S. Tomato Leaf Disease Detection using Convolutional Neural Networkwith Data Augmentation. In Proceedings of the 2020 5th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 10–12 June 2020; pp. 1125–1132.
25. Lintas, A.; Rovetta, S.; Verschure, P.F.; Villa, A.E. Artificial Neural Networks and Machine Learning–ICANN 2017. In Proceedings of the 26th International Conference on Artificial Neural Networks,

Alghero, Italy, 11–14 September2017; Proceedings, Part II; Springer: Cham, Switzerland, 2017; Volume 10614.

26. Kamal, K.C.; Yin, Z.; Wu, M.; Wu, Z. Depthwise separable convolution architectures for plant disease classification.Comput. Electron. Agric. 2019, 165, 104948.
27. Kaluarachchi, T.; Reis, A.; Nanayakkara, S. A Review of Recent Deep Learning Approaches in Human-Centered Machine Learning.Sensors 2021, 21, 2514.
28. Turkoglu M, Yaniko˘glu B, Hanbay D (2021) PlantDiseaseNet: convolutional neural network ensemble for plant disease and pest detection. Signal, Image Video Processing, 1–9
29. Esmael Hamuda, Martin Glavin, and Edward Jones. A survey of image processing techniques for plant extraction and segmentation in the field. 2016. Comput. Electron. Agric. 125, C (July 2016), 184–199.DOI:https://doi.org/10.1016/j.compag.2016.04.024.
30. Batool, A., Hyder, S.B., Rahim, A., Waheed, N., Asghar, M.A.: Fawad: Classification and identification of tomatoleaf disease using deep neural network. In: 2020 Int. Conf. Eng. Emerg. Technol. ICEET 2020 (2020)
31. Kaya, A., Keceli, A.S., Catal, C., Yalic, H.Y., Temucin, H., Tekinerdogan, B.: Analysis of transfer learning for deep neural network based plant classification models. Comput. Electron. Agric. 158(January), 20–29 (2019)
32. E. C. Too, L. Yujian, S. Njuki, and L. Yingchun, "A comparative study of fine-tuning deep learning models for plantdisease identification," Computers and Electronics in Agriculture, vol. 161, pp. 272-279, 2019.

Figures

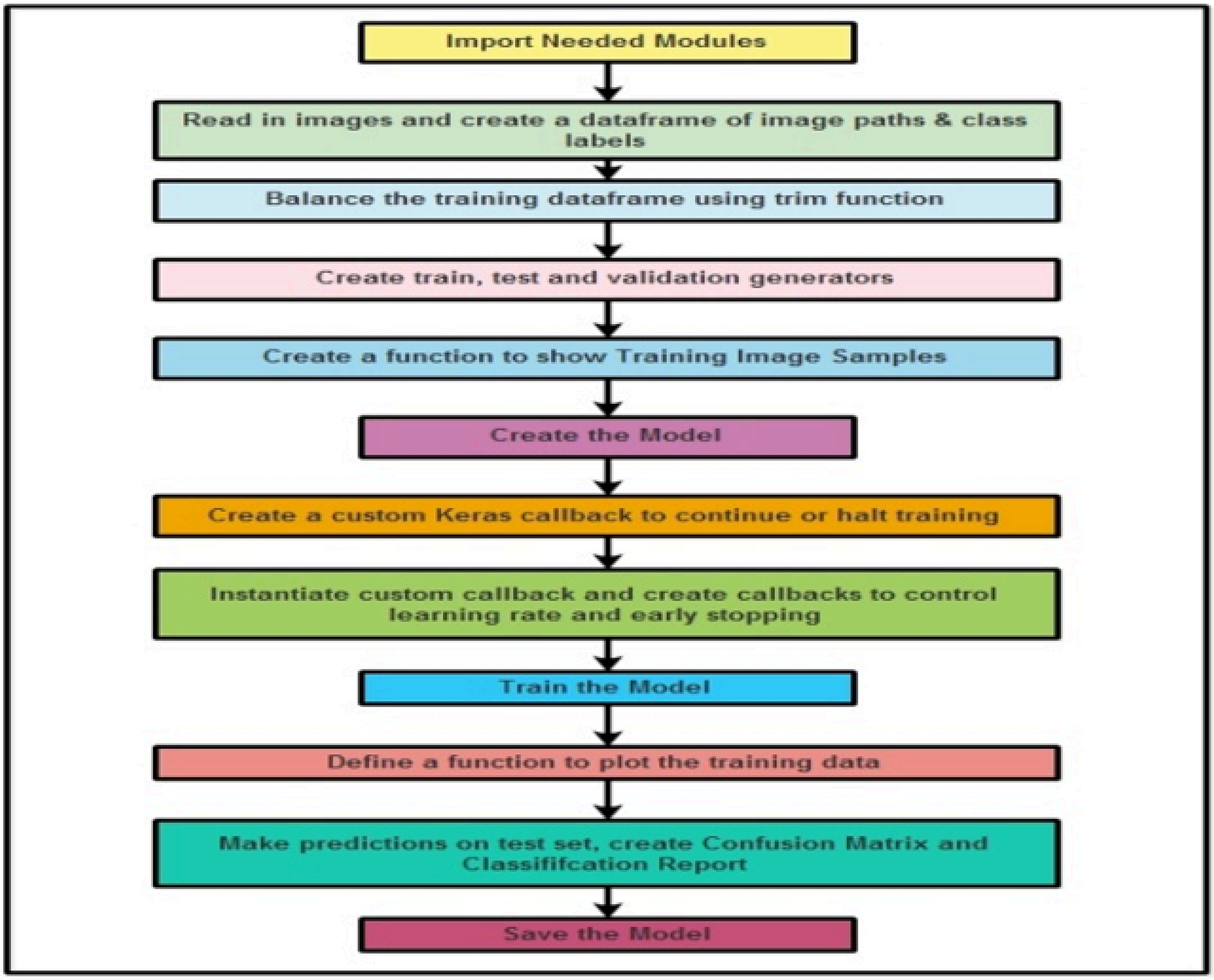


Figure 1

Flowchart of Proposed Model

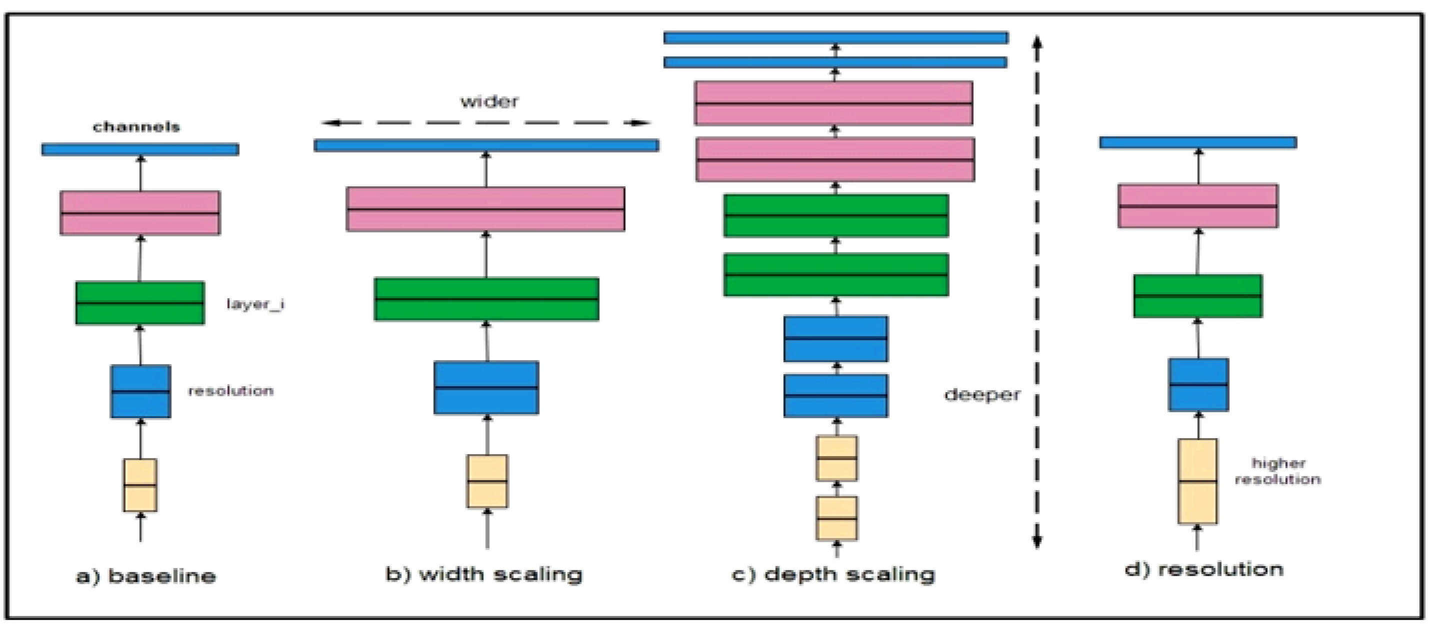


Figure 2

Model Scaling

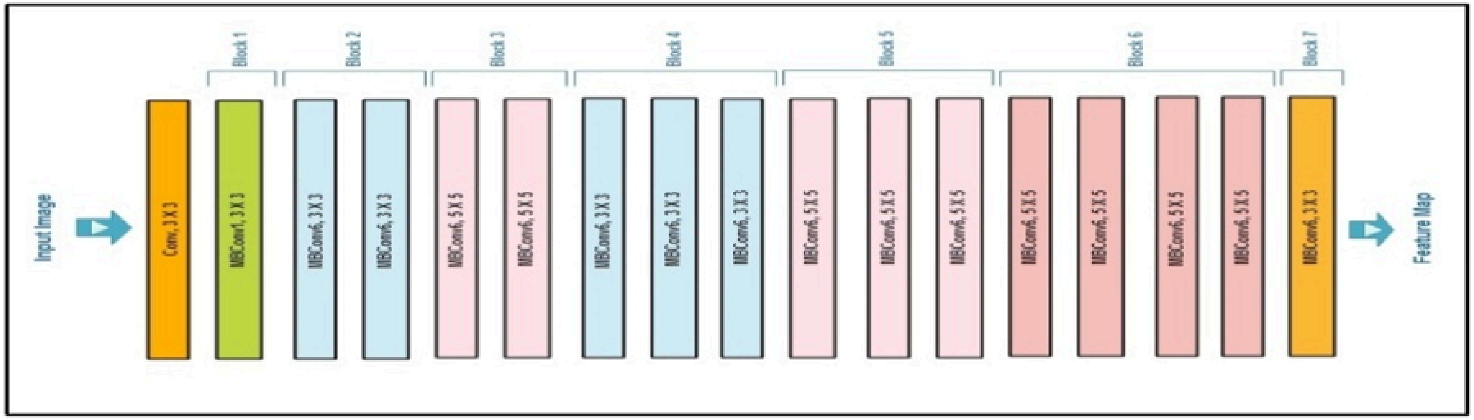


Figure 3

Architecture of EfficientNet

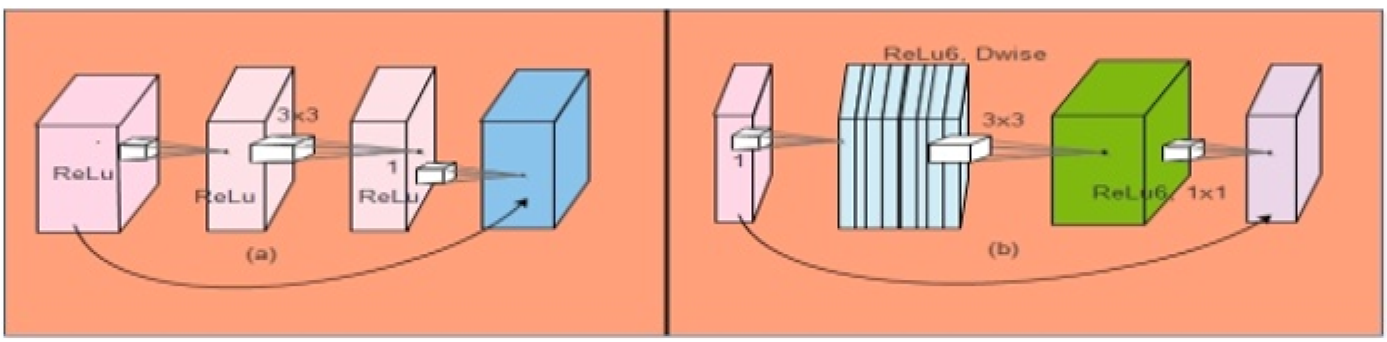


Figure 4

Structure of (a) normal residual block (b) Inverted residual block

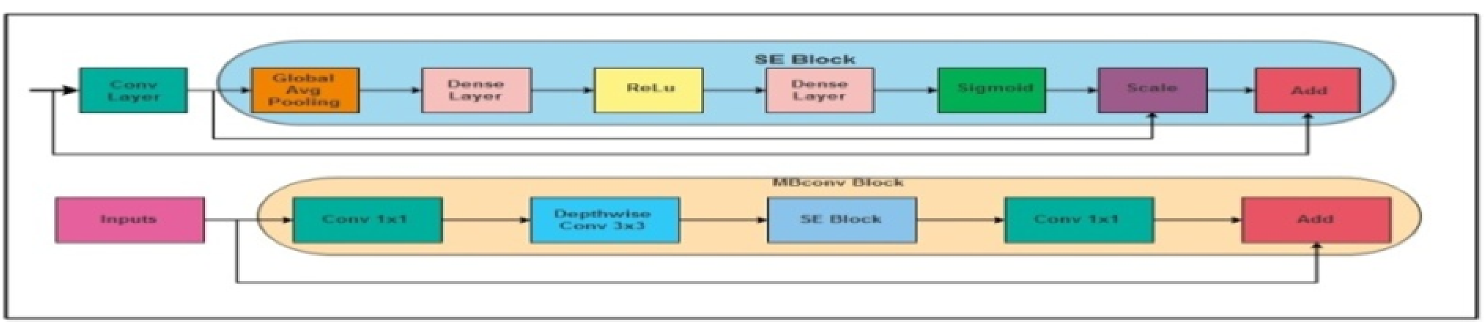


Figure 5

Architecture of an SE & MBConv Block

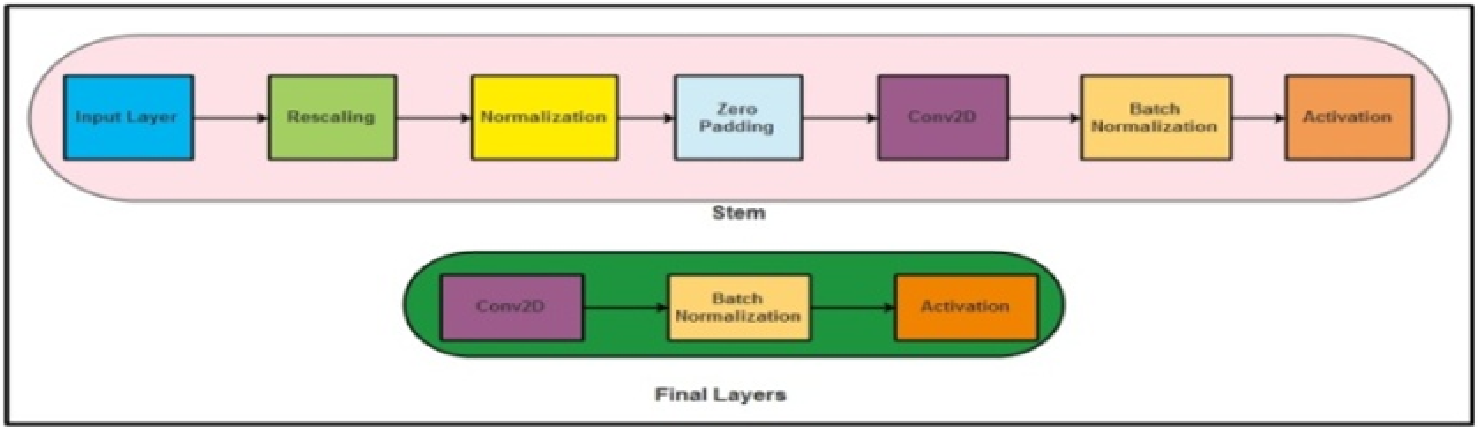


Figure 6

Model Design

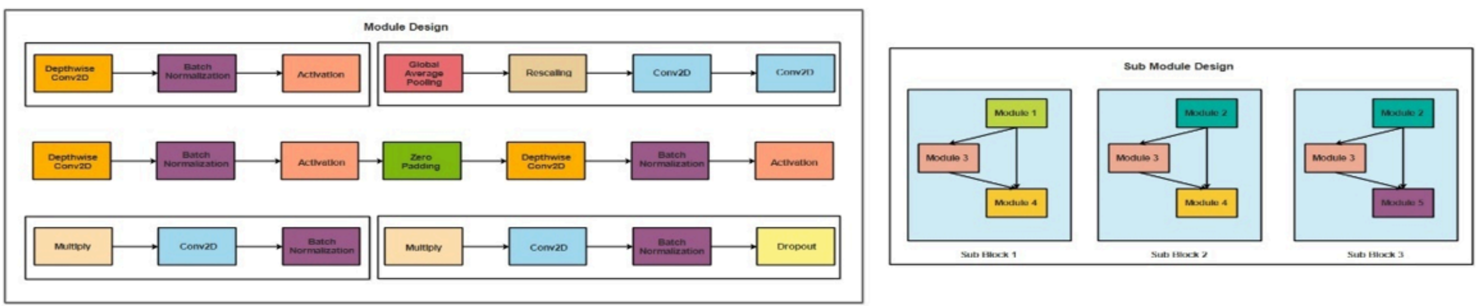


Figure 7

Module& Sub Module Design

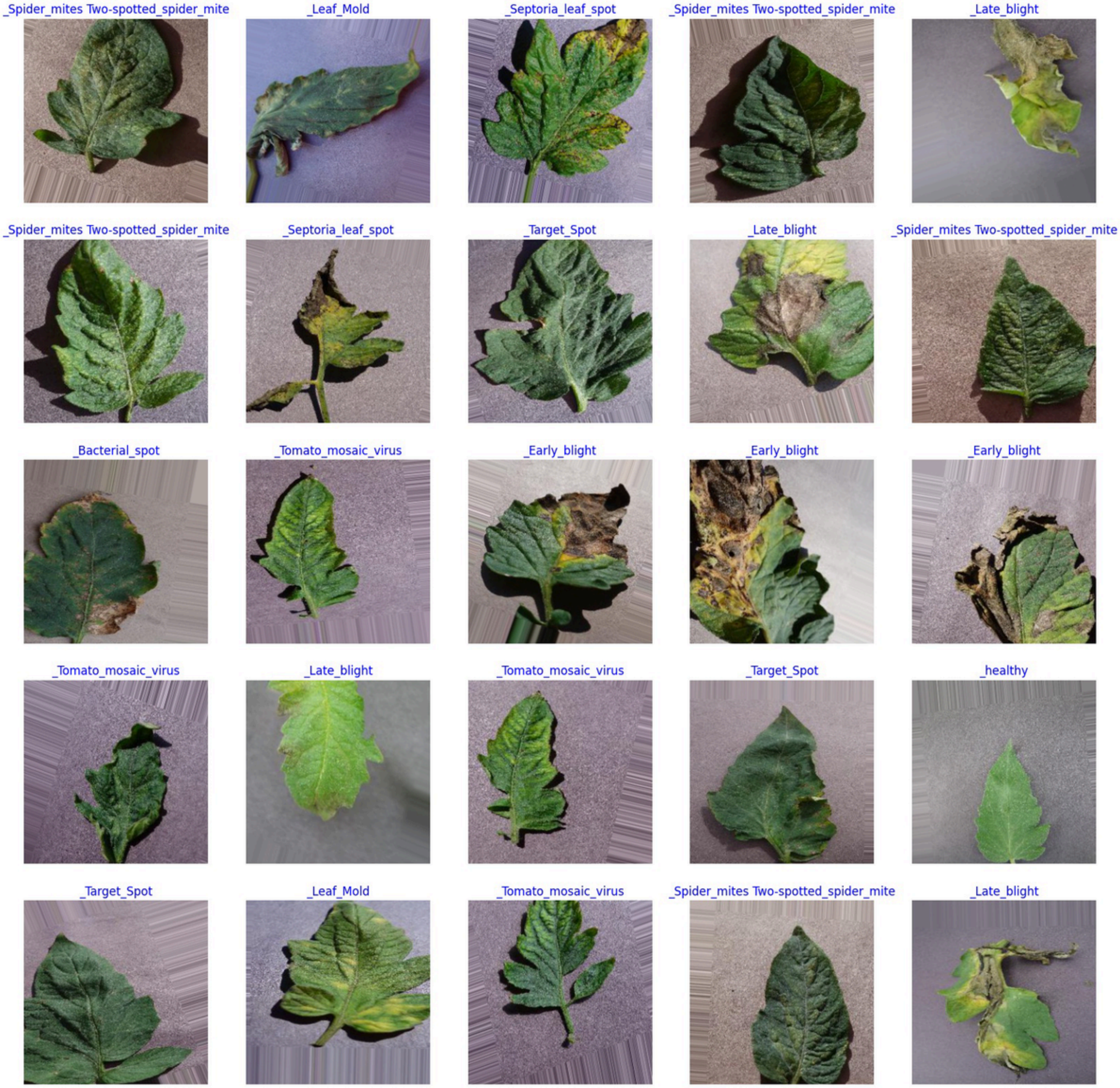


Figure 8

Sample Image of Training Data

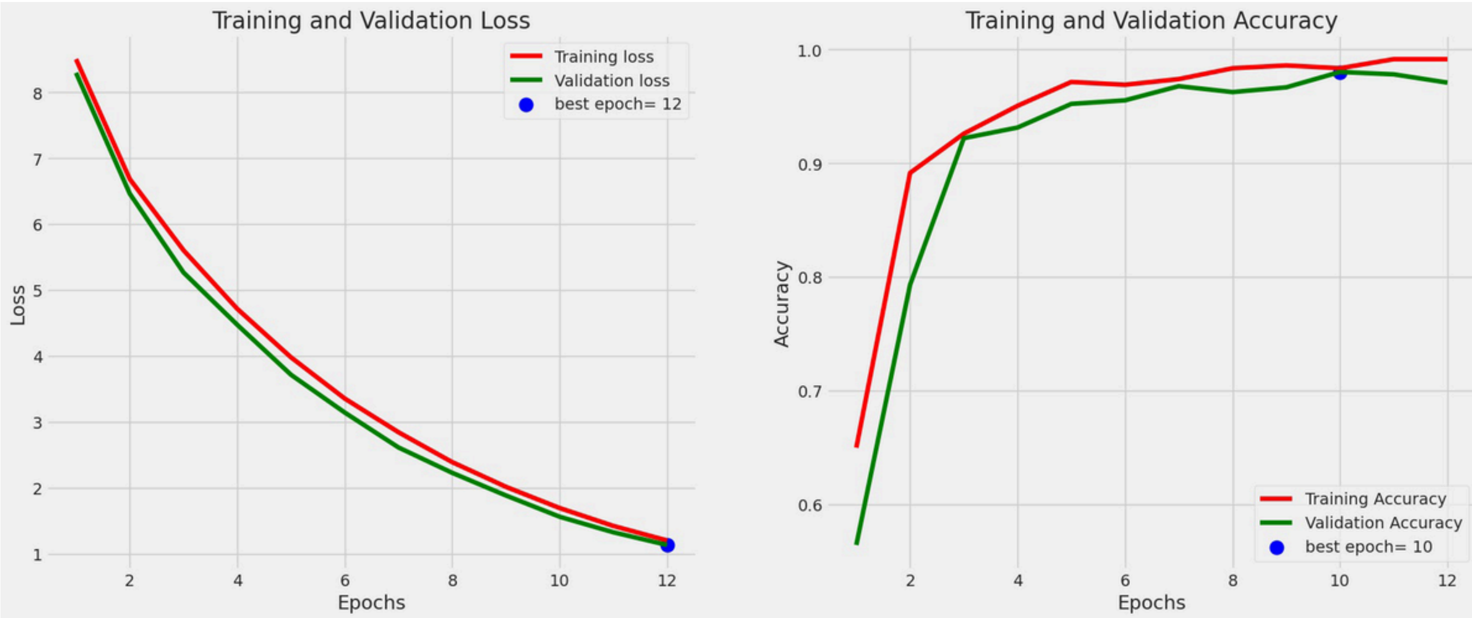


Figure 9

Plot of Training 2000 disease Data with Loss & Accuracy

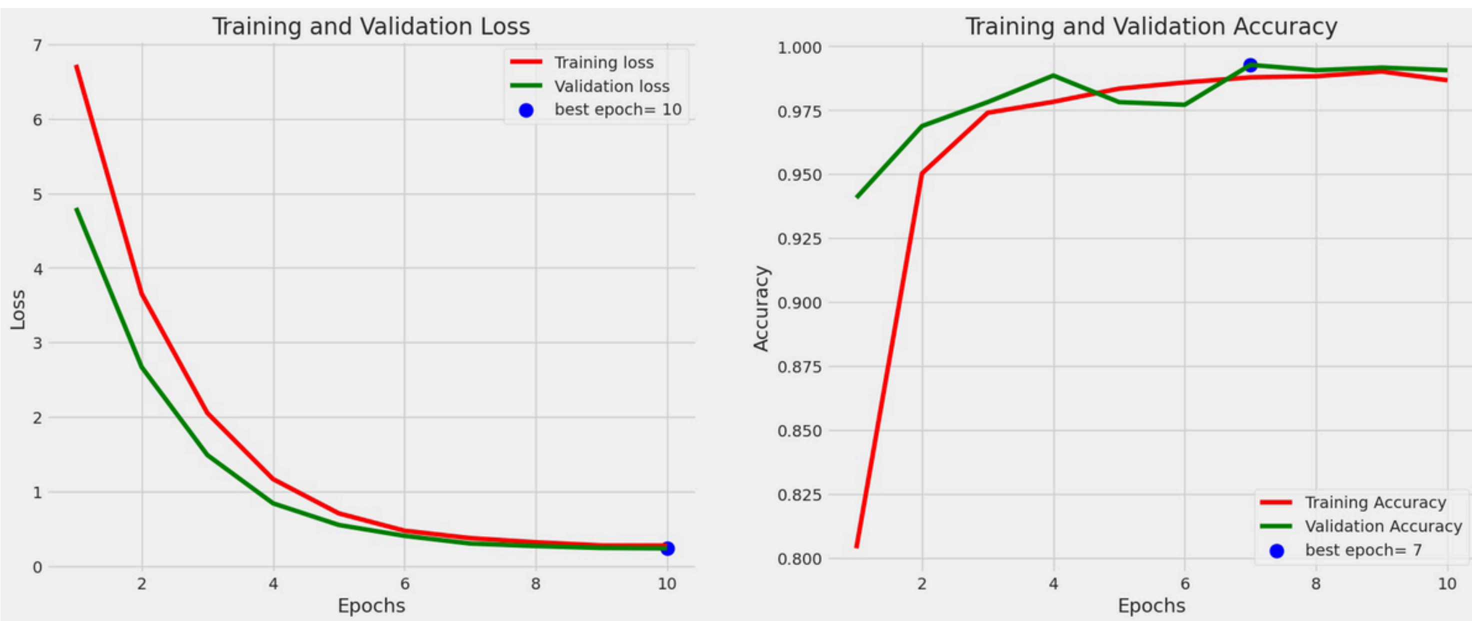


Figure 10

Plot of Training 7000 disease Data with Loss & Accuracy

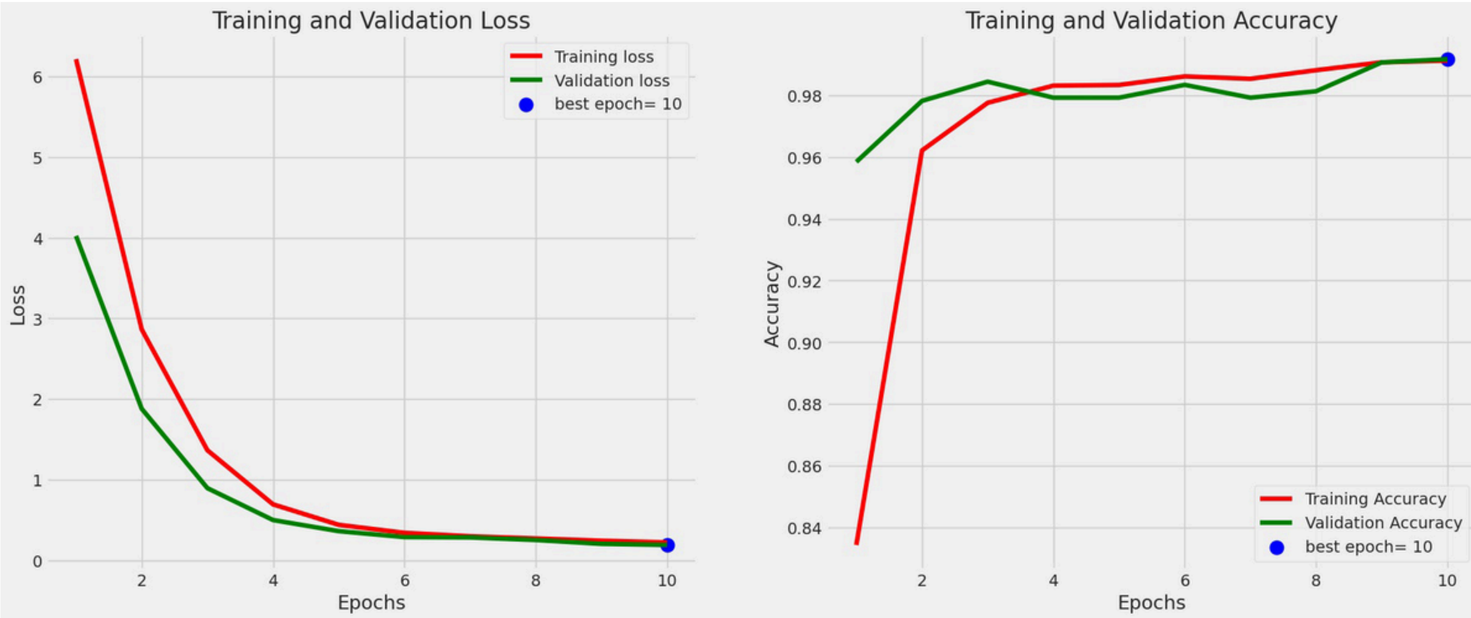
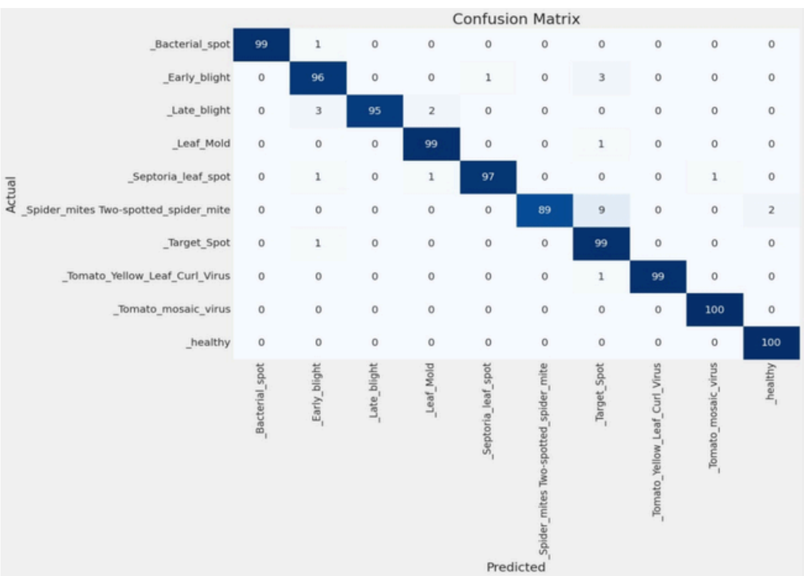


Figure 11

Plot of Training 9000 disease Data with Loss & Accuracy

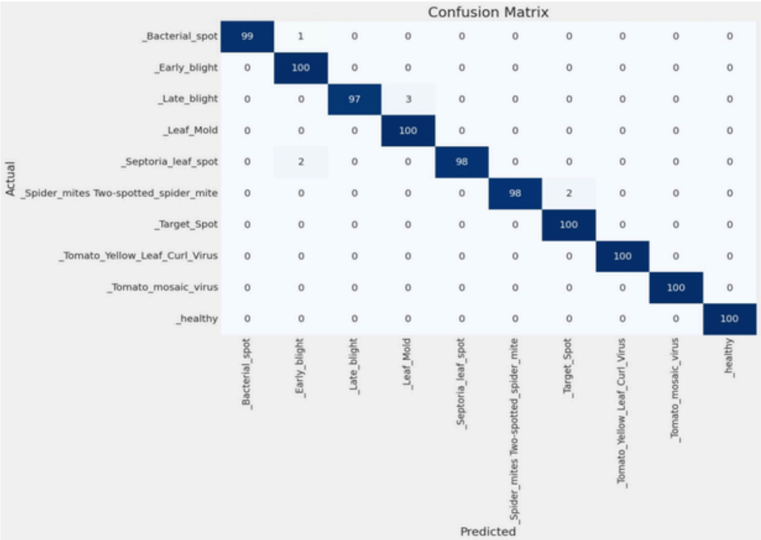


Classification Report:
----

	precision	recall	f1-score	support
_Bacterial_spot	1.0000	0.9900	0.9950	100
_Early_blight	0.9412	0.9600	0.9505	100
_Late_blight	1.0000	0.9500	0.9744	100
_Leaf_Mold	0.9706	0.9900	0.9802	100
_Septoria_leaf_spot	0.9898	0.9700	0.9798	100
_Spider_mites Two-spotted_spider_mite	1.0000	0.8900	0.9418	100
_Target_Spot	0.8761	0.9900	0.9296	100
_Tomato_Yellow_Leaf_Curl_Virus	1.0000	0.9900	0.9950	100
_Tomato_mosaic_virus	0.9901	1.0000	0.9950	100
_healthy	0.9804	1.0000	0.9901	100
accuracy			0.9730	1000
macro avg	0.9748	0.9730	0.9731	1000
weighted avg	0.9748	0.9730	0.9731	1000

Figure 12

Confusion Matrix& Classification Report for 2000 images



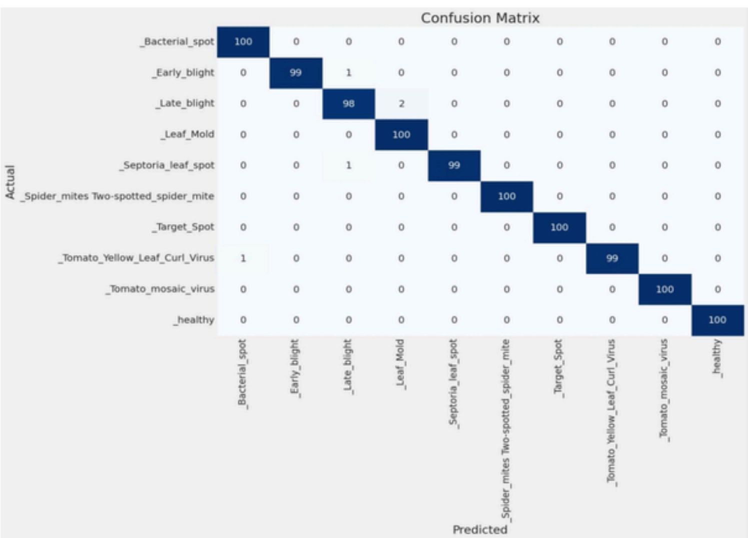
Classification Report:

----

	precision	recall	f1-score	support
_Bacterial_spot	1.0000	0.9900	0.9950	100
_Early_blight	0.9709	1.0000	0.9852	100
_Late_blight	1.0000	0.9700	0.9848	100
_Leaf_Mold	0.9709	1.0000	0.9852	100
_Septoria_leaf_spot	1.0000	0.9800	0.9899	100
_Spider_mites Two-spotted_spider_mite	1.0000	0.9800	0.9899	100
_Target_Spot	0.9804	1.0000	0.9901	100
_Tomato_Yellow_Leaf_Curl_Virus	1.0000	1.0000	1.0000	100
_Tomato_mosaic_virus	1.0000	1.0000	1.0000	100
_healthy	1.0000	1.0000	1.0000	100
accuracy			0.9920	1000
macro avg	0.9922	0.9920	0.9920	1000
weighted avg	0.9922	0.9920	0.9920	1000

Figure 13

Confusion Matrix& Classification Report for 7000 images



Classification Report:

----

	precision	recall	f1-score	support
_Bacterial_spot	0.9901	1.0000	0.9950	100
_Early_blight	1.0000	0.9900	0.9950	100
_Late_blight	0.9800	0.9800	0.9800	100
_Leaf_Mold	0.9804	1.0000	0.9901	100
_Septoria_leaf_spot	1.0000	0.9900	0.9950	100
_Spider_mites Two-spotted_spider_mite	1.0000	1.0000	1.0000	100
_Target_Spot	1.0000	1.0000	1.0000	100
_Tomato_Yellow_Leaf_Curl_Virus	1.0000	0.9900	0.9950	100
_Tomato_mosaic_virus	1.0000	1.0000	1.0000	100
_healthy	1.0000	1.0000	1.0000	100
accuracy			0.9950	1000
macro avg	0.9950	0.9950	0.9950	1000
weighted avg	0.9950	0.9950	0.9950	1000

Figure 14

Confusion Matrix& Classification Report for 9000 images

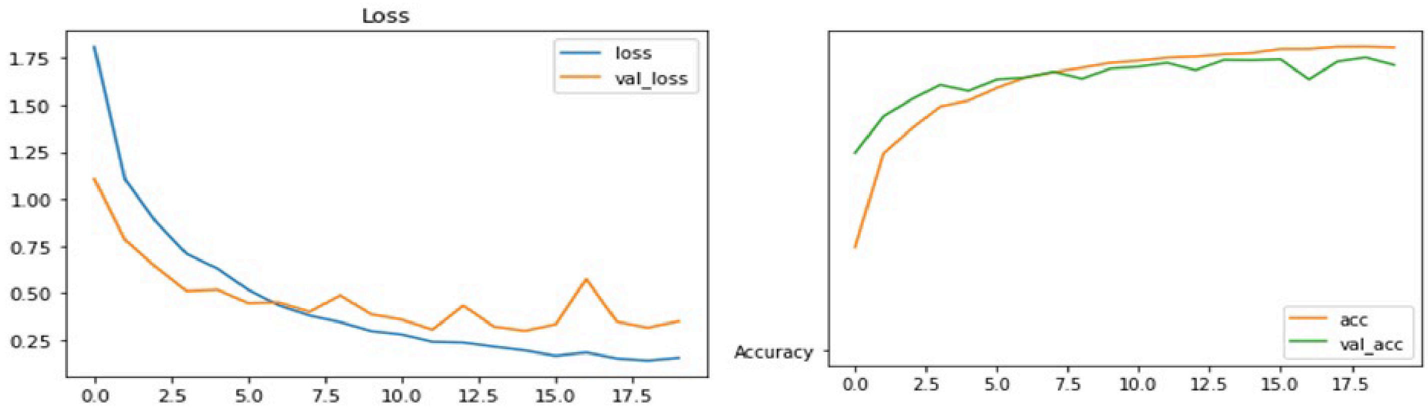


Figure 15

CNN Performance