IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⵗ Peer-reviewed & Refereed journal ⵗ Vol. 14, Issue 7, July 2025

DOI: 10.17148/IJARCCE.2025.14740

TOMATO LEAF DISEASE DETECTION

Pankaj Kumar Gupt[1], Dr. Anita Pal[2]

Scholar M-Tech Final Year, Department of Computer Science & Engineering,

Goel Institute of Technology & Management, Lucknow, India[1]

Associate Professor, Department of Computer Science & Engineering,

Goel Institute of Technology & Management, Lucknow, Uttar Pradesh, India[2]

Abstract: Tomato cultivation is susceptible to various diseases, leading to significant yield loss and economic impact. Rapid and accurate prediction is essential for timely intervention and mitigation. Deep learning techniques, specifically Convolutional Neural Networks (CNN), are applied for automated detection of tomato leaf diseases. The methodology involves acquiring high-resolution images of tomato leaves and training a CNN model to classify them into healthy or diseased categories. The dataset comprises labeled images representing Early Blight, Late Blight, and healthy leaves. The CNN architecture is optimized to achieve high accuracy, precision, recall, and F1-score. The trained model demonstrates promising results in identifying tomato leaf diseases even under environmental variations and leaf deformities. The approach also allows for near real-time detection, enabling timely agricultural interventions. This research contributes to automated agricultural monitoring systems, aiding farmers in early disease detection and management, thereby enhancing crop productivity and sustainability.

Keywords: Tomato Leaf Disease Detection, Convolutional Neural Network (CNN), Deep Learning, Image Classification, Early Blight, Late Blight, Real-time Detection, Precision Agriculture

I. INTRODUCTION

Tomatoes are a vital agricultural crop globally, especially in India, contributing significantly to food security and economic stability. However, tomato leaf diseases caused by pathogens such as fungi, bacteria, and viruses threaten crop health, reduce yield, and cause substantial financial losses. Traditional detection methods rely on manual visual inspection, which is time-consuming, subjective, and prone to human error.

Advances in deep learning and computer vision have revolutionized plant disease detection by enabling automated, accurate, and scalable solutions. Convolutional Neural Networks (CNN) have emerged as a powerful approach for image classification tasks, including the identification of tomato leaf diseases. This study proposes a CNN-based framework for automated classification of tomato leaves into healthy or diseased categories, aiming to improve early detection, reduce crop losses, and promote sustainable farming practices.

1.1 Deep Learning

Training artificial neural networks to learn from and generate predictions from data is the main goal of the machine learning subfield known as "deep learning." Deep learning models automatically learn to extract pertinent features from raw data, in contrast to typical machine learning techniques that could need manual feature extraction. Deep neural networks, which are made up of several layers of connected nodes, or neurons, are used to do this. The model can acquire more intricate representations of the input as each layer processes the data before passing it on to the one after it.

Deep learning has achieved notable success in a number of domains, such as speech recognition, computer vision, and natural language processing. Language translation, object identification, and image classification have all advanced as a result of its capacity to automatically learn hierarchical representations from massive datasets. In order to minimize the discrepancy between expected and actual outputs, deep learning models are trained using optimization techniques like stochastic gradient descent, in which the model's parameters are changed iteratively. All things considered, deep learning has transformed artificial intelligence by allowing robots to carry out tasks that were previously believed to be solely human.

1.2 Applications of Deep Learning

Deep learning can extract complex patterns and representations from vast volumes of data, as illustrated in Fig.1.1, it has a broad range of applications across multiple fields.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁒ Peer-reviewed & Refereed journal ⁒ Vol. 14, Issue 7, July 2025

DOI: 10.17148/IJARCCE.2025.14740

- **Finance:** Deep learning methods are used in the finance industry for tasks like credit scoring, algorithmic trading, fraud detection, risk assessment, and customer relationship management. These methods help financial organizations successfully manage risks and make data-driven decisions.

- **Recommendation Systems:** Online platforms including social media, streaming services, and ecommerce websites use recommendation systems that are powered by deep learning models. In order to recommend appropriate goods, films, or other material, these systems examine user behavior and preferences.

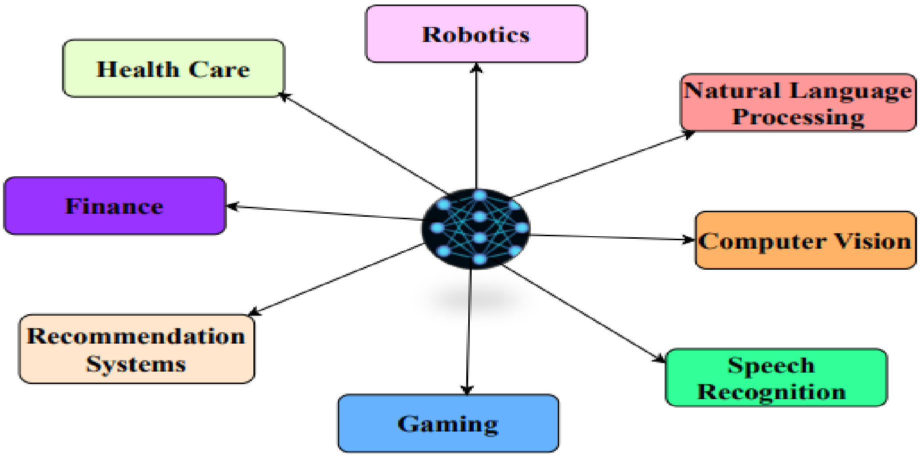

Fig. 1.1 Deep Learning Applications

II. METHODS AND MATERIAL

A. Dataset Acquisition

The dataset consists of high-resolution images of tomato leaves obtained from publicly available sources such as the PlantVillage dataset and field-collected images. The dataset includes three classes: Early Blight, Late Blight, and healthy leaves. the dataset and the total number of images used for training and testing, Table 1 displays the various models used to identify plant diseases

B. Preprocessing

Images undergo preprocessing steps including resizing, normalization, and augmentation (rotation, flipping, brightness adjustment) to improve model robustness.

C. Model Architecture

A CNN model is developed with multiple convolutional and pooling layers, followed by fully connected layers and a softmax output layer. Hyperparameters are tuned to optimize performance. As illustrated in Fig. 2.1, this technique is extremely useful for agriculture as it provides a quick and accurate way to monitor plant health and reduce crop losses (Durmus et al. 2017, Bao et al. 2017).

D. Training and Evaluation

The model is trained using an 80:20 train-test split. Evaluation metrics include accuracy, precision, recall, and F1-score. Comparative analysis is conducted against other architectures such as ResNet-50, DenseNet-121, and VGG-19.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ※ Peer-reviewed & Refereed journal ※ Vol. 14, Issue 7, July 2025

DOI: 10.17148/IJARCCE.2025.14740

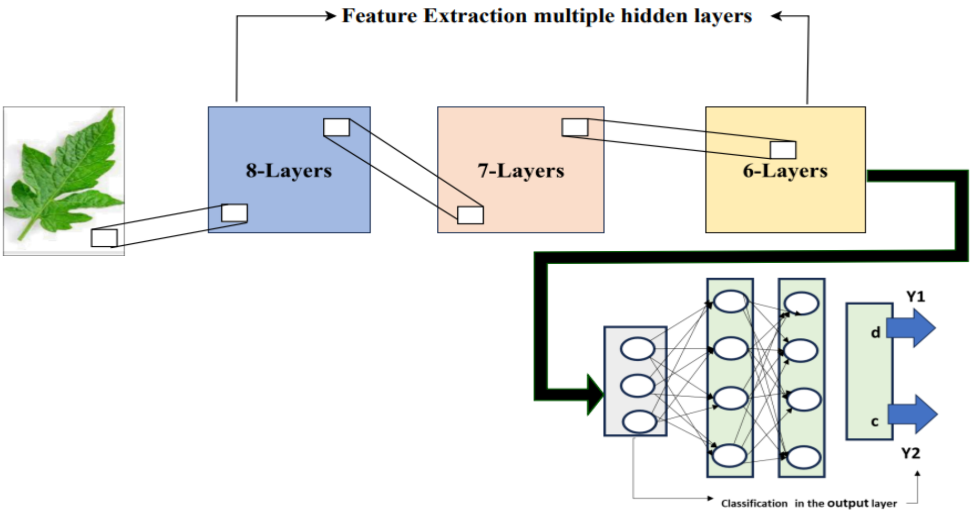

Fig. 2.1: CNN Architecture

Table 1: Dataset Summary

Class	Number of Images	Training Images	Testing Images	Source
Early Blight	3,000	2,400	600	PlantVillage
Late Blight	2,800	2,240	560	PlantVillage
Healthy	3,200	2,560	640	Field & Public
Septoria Leaf Spot	2,500	2,000	500	PlantVillage
Tomato Mosaic Virus	1,800	1,440	360	Field & Public
Bacterial Spot	2,200	1,760	440	PlantVillage
Target Spot	1,900	1,520	380	PlantVillage
Leaf Mold	2,100	1,680	420	PlantVillage

III. RESULTS AND DISCUSSION

The DenseNet-121 model achieved superior performance across all metrics, followed by ResNet-50 and VGG-19. Table 1 summarizes the performance of different architectures The experimental findings and comparison analysis show that, while using fewer epochs than the other models.

Based on an 80/20 split of the dataset and the total number of photos used for training and testing, Table 2 displays the various models used to identify plant diseases along with the appropriate accuracy percentages.

Table 2: Accuracy of Various CNN Models (80:20 Split)

S.N.	Models	Accuracy (80 : 20)	Total Images
1	LeNet	98.0	18378
2	DenseNet Xception	97.10	41263
3	DenseNet-121	99.69	14529
4	MobileNet V1 with Adam optimization	99.0	1432
5	Learning Vector Quantization	90.0	500
6	Resnet-50	98.0	1000
7	CNN	95.0	16011
8	VGG-19	97.0	16000
9	Inception V3 model	95.85	3362

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁘ Peer-reviewed & Refereed journal ⁘ Vol. 14, Issue 7, July 2025

DOI: 10.17148/IJARCCE.2025.14740

Implementation of DenseNet-121

In order to address the vanishing gradient issue DenseNet-121 employs "dense connectivity," a skip connection architecture in which every layer is connected to every subsequent level. This makes it easier for gradients to move throughout the network, which helps the network learn from data. In order to address the vanishing gradient problem, DenseNet-121 used a variety of techniques in addition to dense connections. One technique that helps stabilize training is batch normalization, which standardizes a inputs to every layer. The ReLU activation function, which ensures that the gradients are nonzero for positive inputs, can also help with the vanishing gradient issue.

As seen in Figs. 3.1, 3.2, 3.3, and 3.4, both ResNet-101 and VGGNet were installed, and their accuracy performances on pretrained networks were evaluated using 50 and 80 epochs and for different examples. The accuracies attained by VGGNet, ResNet-101, and ResNet-152 are 94.67%, 95.87%, and 96.97%, respectively. Because of their depth and quantity of parameters, ResNet-101 and ResNet-152 have high computing requirements. It is crucial to remember that VGGNet can be extremely computationally intensive, particularly in its more complex versions. It is more challenging to assess the learnt characteristics and their relationship to disease patterns in tomato leaves due to the increased complexity of deeper architectures like ResNet-152, ResNet-101, and VGGNet. The ability to use pre-trained models makes these designs desirable. These pre-trained models can be used and improved on particular tomato leaf datasets on diseases using transfer learning, saving time and effort in contrast to starting from zero when exercising. However, ResNet-152 has issues with disappearing gradients.

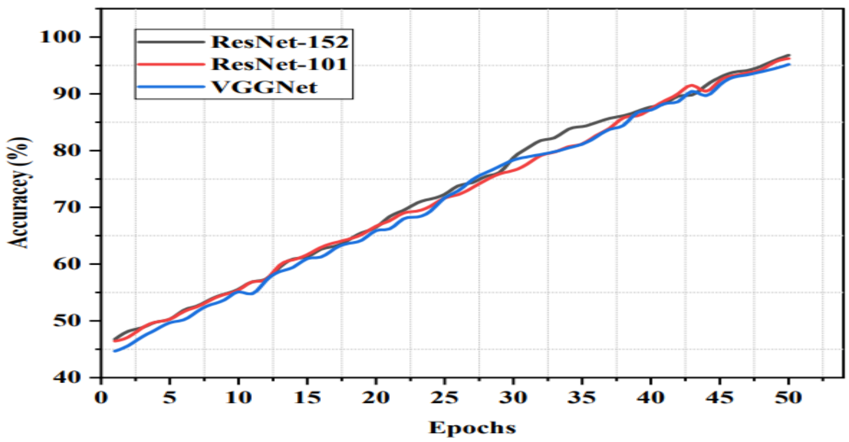

Figure 3.1: Instance I with 50-epochs.

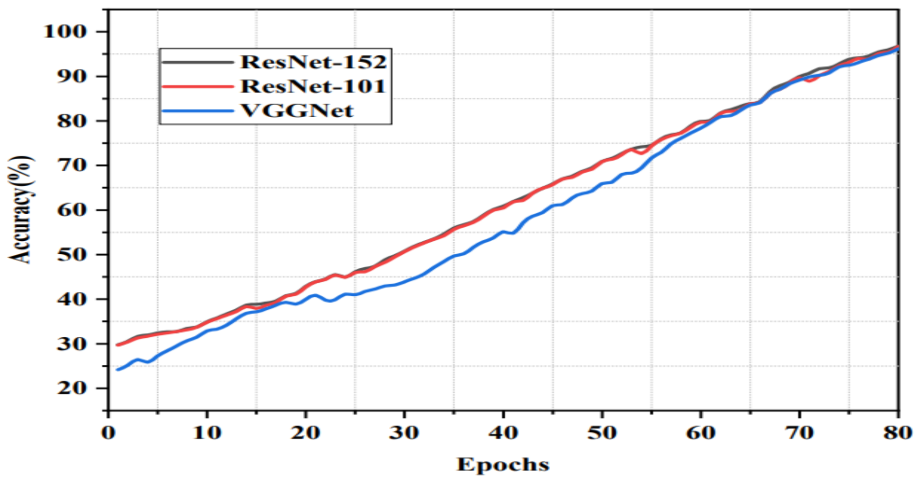

Figure 3.2: Instance-I with 80-epochs.

To train the model, the DenseNet-121 architecture used a dataset that included pictures of both healthy and injured tomato leaves. This dataset has 18,160 photos that have been categorized appropriately for the training procedure, which uses a categorical cross-entropy loss function and 50 iterations with a batch size of 32. With the learning rate set to 0.001, the

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⪾ Peer-reviewed & Refereed journal ⪾ Vol. 14, Issue 7, July 2025

DOI: 10.17148/IJARCCE.2025.14740

Adam optimizer was employed for optimization. At each layer of the neural network, batch normalization is used to normalize the inputs by subtracting the mean and dividing by the standard deviation in order to improve training stability. It has been demonstrated that by lowering the internal covariate shift, this method enhances the model's.

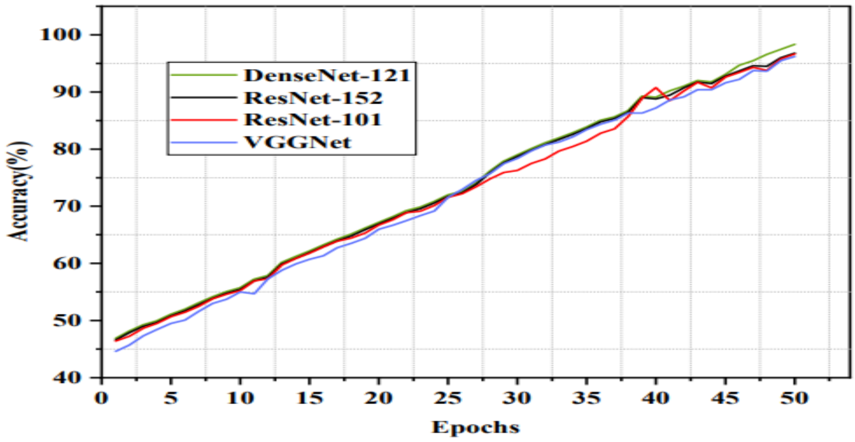

Figure 3.3: Instance-II with 50-epochs.s

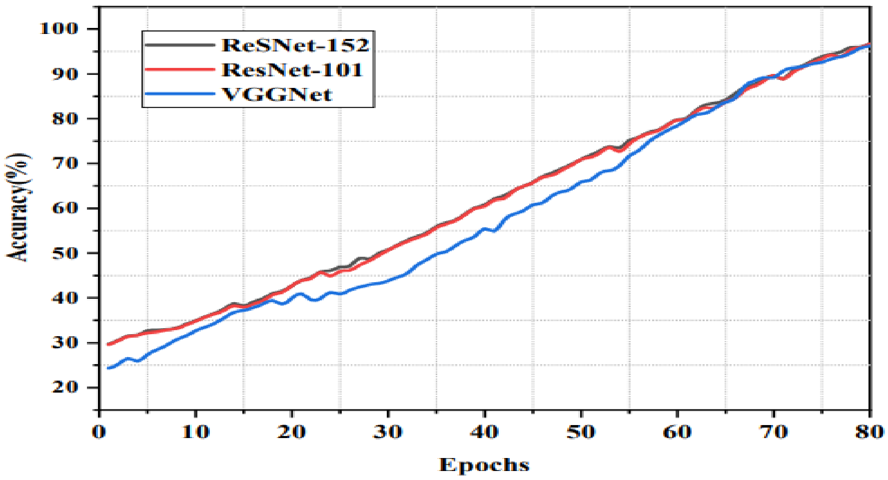

Figure 3.4: Instance-II with 80-epochs.

A thick connection method that merges the current layer's output with the output of the previous layer , allows the gradients to move from later layers to earlier layers. After five epochs of no improvement, the training process is abruptly stopped to avoid overfitting.

A callback function is used to store the model with the highest validation accuracy. F1-score, accuracy, precision, and recall were among the measures used to evaluate the model's performance on the test dataset. Furthermore, a performance comparison of the proposed model with existing tomato leaf disease detection techniques was conducted.

A 70:30 split of the data used to train the DenseNet-121 model on the PlantVillage dataset was used to assess how well the proposed method identified illnesses in tomato plant leaves. When compared to state-of-the-art methods for rapidly identifying tomato leaf disease, the results demonstrate that the proposed method is more accurate than the alternatives. The model's overall accuracy was 98.3%. The efficiency of the proposed method for detecting tomato leaf diseases using the DenseNet-121 architecture to create an evaluation parameter table. An example table of evaluation parameters is shown in Table demonstrating stable convergence.

Early disease detection enables timely intervention, reducing crop losses and minimizing pesticide use. Integration with mobile and web platforms enables real-time disease detection in the field, supporting precision agriculture and sustainable farming practices.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 7, July 2025
DOI: 10.17148/IJARCCE.2025.14740

IV. CONCLUSION

This research demonstrates that CNN-based deep learning models can accurately detect and classify tomato leaf diseases. The DenseNet-121 architecture achieved 99.69% accuracy, proving effective for real-world agricultural applications. Future work will focus on integrating temporal models such as LSTM for predicting disease progression and expanding the dataset to include more disease types and environmental conditions.

REFERENCES

[1]. S. Durmus, E. O. Gunes, and M. Kirci, "Disease detection on the leaves of the tomato plants by using deep learning," 2017 6th International Conference on Agro-Geoinformatics, 2017.
[2]. M. A. Ramcharan, P. McCloskey, and J. Baranowski, "Deep learning for image-based plant disease detection," Frontiers in Plant Science, 2019.
[3]. K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Proceedings of the IEEE CVPR, 2016.
[4]. PlantVillage Dataset. [Online]. Available: https://plantvillage.psu.edu
A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," NIPS, 2012.
[5]. F. Chollet, "Xception: Deep learning with depthwise separable convolutions," Proceedings of the IEEE CVPR, 2017.
[6]. G. Huang, Z. Liu, L. Van Der Maaten, and K. Weinberger, "Densely connected convolutional networks," Proceedings of the IEEE CVPR, 2017.
[7]. M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," Proceedings of the IEEE CVPR, 2018.
[8]. S. Albawi, T. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," 2017 International Conference on Engineering and Technology (ICET), 2017.
[9]. R. Yamashita, M. Nishio, R. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," Insights into Imaging, 2018.
[10]. D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.
[11]. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, pp. 436–444, 2015.
A. Szegedy, W. Liu, Y. Jia, P. Sermanet et al., "Going deeper with convolutions," Proceedings of the IEEE CVPR, 2015.
[12]. T. Y. Lin, P. Dollár, R. Girshick et al., "Feature pyramid networks for object detection," Proceedings of the IEEE CVPR, 2017.
[13]. J. Deng, W. Dong, R. Socher et al., "ImageNet: A large-scale hierarchical image database," Proceedings of the IEEE CVPR, 2009.
[14]. M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes Challenge (VOC)," International Journal of Computer Vision, 2010.
[15]. P. Mohanty, D. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," Frontiers in Plant Science, 2016.
[16]. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," Proceedings of the IEEE CVPR, 2015.
[17]. W. Liu, D. Anguelov, D. Erhan et al., "SSD: Single shot multibox detector," Proceedings of the European Conference on Computer Vision (ECCV), 2016.
[18]. R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," Proceedings of the IEEE CVPR, 2014.
[19]. M. Tan and Q. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," Proceedings of the International Conference on Machine Learning (ICML), 2019.
[20]. J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," ACL, 2018.
[21]. H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," Proceedings of the International Conference on Machine Learning (ICML), 2021.
A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit et al., "Attention is all you need," Proceedings of the NeurIPS, 2017.
[22]. X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," Proceedings of the IEEE CVPR, 2018.
[23]. S. J. Pan and Q. Yang, "A survey on transfer learning," IEEE Transactions on Knowledge and Data Engineering, 2010.

