Research Article

AI-Driven Detection of Tomato Leaf Diseases for Sustainable Agriculture

Swetha R. Kumar*
School of Electrical Engineering, Vellore Institute of Technology, Chennai, India

Mogana Priya Chinnasamy
School of Mechanical Engineering, Vellore Institute of Technology, Chennai, India

* Corresponding author. E-mail: swetha.rkumar@vit.ac.in DOI: 10.14416/j.asep.2025.06.007
Received: 13 March 2025; Revised: 21 April 2025; Accepted: 15 May 2025; Published online: 24 June 2025

Abstract
This study explores a novel approach for detecting diseases in tomato leaves through the application of neural networks, aiming to enhance early diagnosis and management strategies for farmers and plant pathologists. The research investigates nine prevalent diseases affecting tomato foliage, including Early Blight, Late Blight, Septoria Leaf Spot, Target Spot, Yellow Leaf Curl Virus, Bacterial Spot, Spider Mites, Leaf Mold, Tomato Mosaic Virus, and Healthy leaves, using pre-trained deep learning models, ResNet-34 and VGG16. A diverse dataset of tomato leaf images, exhibiting various disease symptoms under field and curated conditions, was pre-processed, labeled, and split into training (80%) and testing (20%) sets to fine-tune the models. Evaluation of the testing dataset revealed that ResNet-34 achieved a higher accuracy of 99% compared to VGG16's 89%, demonstrating superior performance in disease classification. Precision, recall, and F1 scores further confirmed ResNet-34's robustness, averaging 0.99 across classes. These findings highlight the efficacy of deep learning in agricultural disease detection, contributing to sustainable practices by enabling timely interventions, reducing crop losses, and minimizing pesticide use. The study underscores the potential of AI-driven solutions to transform tomato cultivation, paving the way for scalable, real-time applications in resource-constrained farming environments.

Keywords: Deep learning, Disease classification, Neural networks, Sustainable agriculture, Tomato leaf diseases

1 Introduction

Tomatoes are a vital crop in global agriculture, which contributes significantly to food security and economic constancy. However, tomato cultivation is highly susceptible to various diseases that adversely affect yield quality and quantity. Bacteria, fungi, viruses and other pathogens often go undetected until they have caused severe damage. Early diagnosis and management of such diseases are crucial for reducing losses and ensuring sustainable agricultural practices. The traditional methods of identifying plant diseases, such as visual inspections by agriculture experts, are slow, labor-intensive, and often prone to inaccuracies, particularly in large-scale farming. Thus, there has been a growing interest in utilizing advanced technologies such as deep learning to mechanize and enhance the process of disease detection [1]–[3]. AI-driven disease detection enhances sustainability in agriculture by enabling early identification of tomato leaf diseases, which reduces the need for broad-spectrum pesticide applications—a practice that can harm ecosystems and increase costs. By providing precise, timely diagnoses, the system allows farmers to implement targeted interventions, minimizing crop losses and optimizing resource use (e.g., water, fertilizers), thereby supporting long-term food security and environmental health. Neural networks, with their ability to learn and extract intricate patterns from data, have developed as a promising solution to this challenge.

Recent improvements in deep learning and computer vision technologies have led to substantial progress in the automation of plant disease detection.

However, existing studies face challenges in achieving robustness, scalability, and practical applicability in real-world conditions. CNNs were employed to classify 38 diseases across 14 crops using the PlantVillage dataset, achieving high accuracy. However, the dataset consisted mainly of laboratory-controlled images with uniform backgrounds, limiting the model's applicability to field conditions [4]. The application of CNNs was investigated including AlexNet and VGG16, for plant disease detection, showing impressive results. However, the study highlighted inconsistencies in performance when the models were tested on unseen field data with environmental variations such as lighting and background complexity [5]. Transfer learning with data augmentation was applied for tomato disease detection to improve classification performance. Nevertheless, the study primarily focused on accuracy without addressing computational efficiency, making it less feasible for on-field applications [6].

A shallow CNN technique was implemented for tomato disease classification and reported moderate accuracy levels. The model was computationally light but lacked the depth required to capture complex disease patterns [7]. A robust deep learning system was introduced for tomato pest and disease detection using Faster R-CNN. While effective, the approach required extensive computational resources and suffered from high inference times, making it unsuitable for real-time applications [8]. Different CNN architectures were compared, including MobileNet and DenseNet, for plant disease classification. The study noted that deeper networks often achieved better accuracy but were computationally expensive, highlighting the trade-off between model performance and efficiency [9]. The effectiveness of ResNet architectures was evaluated for plant disease detection [10]. While ResNet models demonstrated superior performance in addressing vanishing gradient issues, their use in agriculture remains underexplored for crop-specific diseases like tomato leaf diseases.

A basic CNN model for crop disease detection was used to achieve reasonable accuracy. However, the study lacked a robust evaluation of its generalizability across diverse datasets [11]. Ensemble learning was utilized to combine multiple models for enhanced classification. Although this approach improved accuracy, it introduced computational complexity, making real-time deployment challenging [12]. Data pre-processing techniques were proposed by researchers to enhance CNN-based plant disease detection [13]–[15]. While this improved the model's accuracy, the study did not address the performance of pre-trained architectures for specific crops.

This study aims to develop a robust, AI-driven solution for detecting nine common tomato leaf diseases using deep learning techniques particularly using ResNet-34 and VGG16, addressing the limitations of prior work and promoting sustainable agricultural practices through early and accurate disease management.

2 Materials and Methods

The approach aims to deliver a dependable and helpful tool for farmers and plant pathologists to identify diseases early, enabling timely involvement and increased crop management strategies. Our proposed study addresses these limitations through the following contributions: Unlike previous studies that rely primarily on controlled datasets with uniform backgrounds, this study utilizes a real-world dataset comprising tomato leaf images captured under diverse field conditions. This approach enhances the robustness and scalability of the model, ensuring that it performs well in practical agricultural environments. Additionally, the study goes beyond evaluating individual models by conducting a systematic comparison between ResNet-34 and VGG16 architectures. This comparison helps identify the more effective model for accurately detecting tomato leaf diseases.

Furthermore, the study focuses on disease-specific optimization by fine-tuning both models to detect nine common tomato leaf diseases, rather than using generalized plant disease data. This targeted approach significantly improves diagnostic precision. By combining the computational efficiency of ResNet-34 with the deep feature extraction capabilities of VGG16, the proposed solution achieves a balance between high accuracy and practical deployment. This makes the system suitable for use in resource-constrained settings, such as remote farms or low-tech agricultural operations. By addressing the gaps in existing literature, this research provides a solution for tomato leaf disease detection, contributing to sustainable agricultural practices and improved yield management.

2.1 *Dataset collections*

The dataset for this study was sourced from the Kaggle repository, comprising labeled images of tomato leaves affected by nine common diseases, including Early blight, Late blight, Septoria leaf spot, Target spot, Yellow leaf curl virus, Leaf mold and Tomato mosaic virus, as well as healthy leaves. Images were obtained from diverse environments, including field conditions and curated datasets, ensuring a balanced representation of different disease conditions and environmental variations. This study utilized a pre-existing dataset from Kaggle eliminating the need for a custom imaging setup [16].

The dataset was sourced from diverse environments, including field conditions with varying weather (e.g., overcast, sunny) and image quality (e.g., low resolution, noise), as noted in its documentation. While not all images are field-captured—some are curated—the inclusion of real-world variability enhances model robustness compared to lab-only datasets (e.g., PlantVillage). Kaggle's reliability stems from its community validation and widespread use in plant disease studies, though limitations like potential bias in disease representation were mitigated through preprocessing and balanced class distribution.

The dataset includes diverse images of tomato leaves, as exemplified in Figures 1 through 10, which illustrate characteristic symptoms of the ten target classes. Figures 1 through 10 illustrate the ten target classes in the dataset, showcasing the variability in disease symptoms and environmental conditions. Figure 1 shows Early Blight with concentric ring lesions and yellow halos. Figure 2 depicts Late Blight with dark, water-soaked lesions and necrosis. Figure 3 presents Septoria Leaf Spot with irregular grayish spots and dark borders. Figure 4 displays the Target Spot with concentric lesions and grayish-white centers. Figure 5 illustrates the Tomato Mosaic Virus with mottled yellowing and leaf distortion. Figure 6 shows a bacterial spot with small, dark spots and yellow halos. Figure 7 depicts Spider Mites with stippling and yellowing from mite feeding. Figure 8 presents Leaf Mold with yellowing and chlorosis from fungal infection. Figure 9 illustrates the Yellow Leaf Curl Virus with upward curling and yellowish discoloration. Figure 10 shows Healthy leaves with a uniform green color and no symptoms. These figures highlight the dataset's diversity, necessitating robust preprocessing for model generalizability.

These nine diseases were selected for analysis due to their high prevalence and significant economic impact on tomato cultivation globally. Early Blight, Late Blight, and Bacterial Spot, for instance, are among the most common fungal and bacterial diseases affecting tomato yields, while viral diseases like Tomato Mosaic Virus and Yellow Leaf Curl Virus pose persistent threats due to their rapid spread and difficulty in management. Including a range of fungal, bacterial, viral, and pest-related conditions ensures the study addresses the most critical challenges faced by farmers, enhancing the practical relevance of the proposed solution.

2.2 *Data preprocessing*

Preprocessing for tomato leaf disease detection is an essential step to ensure the dataset is clean, consistent, and ready for analysis [16]. It begins with collecting a diverse set of labeled images representing different diseases and healthy leaves under various lighting conditions and backgrounds. The data is then cleaned by removing corrupt or unreadable images and ensuring labels are consistent across all classes. If needed, images can be annotated to highlight regions of interest, such as disease spots, using tools like Labelling. To prepare the data for deep learning models, all images are resized to a uniform size, such as 256×256 pixels, which is suitable for popular architectures like ResNet or MobileNet. Cropping can also be applied to eliminate unnecessary background and focus on the leaf areas. These preprocessing steps improve the quality of the dataset, enhance model performance, and reduce noise, enabling more accurate detection of tomato leaf diseases.

Figure 1: Sample images of early blight.

Figure 2: Sample images of late blight.

Figure 3: Sample images of septoria leaf spot.

Figure 4: Sample images of target spot.

Figure 5: Sample images of tomato mosaic virus.

Figure 6: Sample images of bacterial spot.

Figure 7: Sample images of spider mites.

Figure 8: Sample images of leaf mold.

Figure 9: Sample images of yellow leaf curl virus.

Figure 10: Sample images of healthy leaves.

2.3 *Model architectures*

This study utilizes two pre-trained deep learning architectures, ResNet-34 and VGG16, known for their robust performance in image classification tasks. These architectures were selected for this study due to their established performance in image classification tasks and their complementary architectural strengths. ResNet-34, with its residual connections, mitigates the vanishing gradient problem, enabling effective training of deeper networks, which is crucial for capturing complex disease patterns. VGG16, known for its simplicity and uniform architecture, excels in feature extraction and transfer learning, making it a reliable baseline for comparison. These models balance accuracy and computational feasibility, making them suitable for agricultural applications where resource constraints are common.

While ResNet-34 and VGG16 are established architectures and tomato disease detection has been explored (e.g., Brahimi *et al*., Fuentes *et al*.,), this study introduces novelty by systematically comparing these models on a diverse, field-representative dataset of 1,678 images, optimizing them for ten tomato-specific target classes, and balancing accuracy with practical efficiency. Unlike Mohanty *et al*., who used controlled images, or Ferentinos, who addressed multiple crops without crop-specific tuning, our approach enhances robustness and applicability for tomato cultivation under real-world variability.

2.3.1 *ResNet-34 architecture*

ResNet-34, a deep convolutional neural network from the ResNet family, introduced in the 2015 publication 'Deep Residual Learning for Image Recognition' [17], addresses the vanishing gradient problem common in deep networks through residual connections. Residual connections, also known as skip connections, are the main enhancement of ResNet-34 as depicted in Figure 11. They enable the network to learn residual functions rather than direct mappings. Deeper architectural training is made possible by these remaining connections, which facilitate the gradient's easier passage through the network during backpropagation.

The 34 layers that make up the ResNet-34 architecture include 3×3 convolutional layers arranged in residual blocks. The network starts with max pooling after a 7×7 convolutional layer with 64 filters and a stride of 2. The architecture is then separated into four phases, each of which has several leftover blocks. With each level, the number of filters doubles (64, 128, 256, and 512 filters), enabling the network to extract features that are more abstract.

Convolutional layers are followed by a fully connected layer with softmax activation and a global average pooling layer to get classification outputs. ResNet-34 performs exceptionally well on big datasets like ImageNet and is especially useful for image recognition tasks.

2.3.2 VGG16 architecture

VGG16, proposed by Simonyan and Zisserman is a deep convolutional neural network noted for its simplicity and effectiveness. The 16 weighted layers in the architecture are mostly composed of 3×3 convolutional filters that have been applied with padding and stride 1 to maintain spatial dimensions as depicted in Figure 12. [18]. Unlike more complex architectures, VGG16 uses a uniform approach to building depth by stacking convolutional layers, which increases the network's capacity to learn hierarchical features. The VGG16 architecture begins with a series of convolutional layers in five blocks, with each block followed by a max pooling layer. The number of filters increases progressively across the blocks (64, 128, 256, and 512), allowing the network to capture more complex features as the depth increases. The network has three fully connected layers after the convolutional blocks. The first two layers have 4096 neurons each, while the third layer is a softmax layer for classification. VGG16 is still a well-liked option for transfer learning and is a standard for many computer vision tasks despite its many parameters [19].

Both ResNet-34 and VGG16 are highly influential architectures in deep learning, each with its strengths. ResNet-34 excels in handling very deep networks, using residual connections to mitigate training challenges and achieve higher efficiency. VGG16 is simpler in design and widely used for feature extraction and transfer learning but has a higher computational cost due to its fully connected layers and uniform filter stacking. These two models were initialized with pre-trained weights from the ImageNet dataset and adjusted using the tomato leaf disease dataset. A comparative study of both architectures is listed in Table 1.

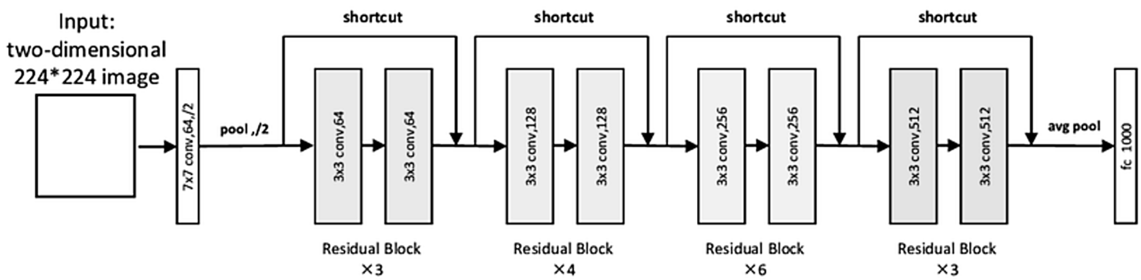

Figure 11: ResNet-34 architecture.

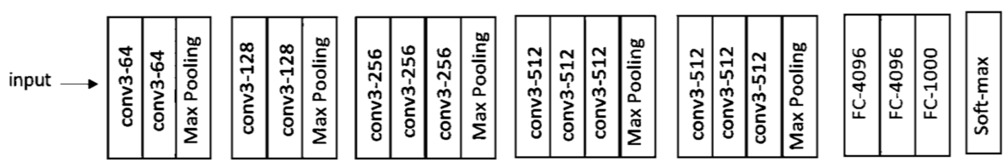

Figure 12: VGG16 architecture.

Table 1: Comparison of ResNet-34 and VGG16 architectures.

Parameter	ResNet-34	VGG16
Year Introduced	2015 (ResNet family by He *et al.)*	2014 (by Simonyan and Zisserman)
Primary Innovation	Residual connections to solve vanishing gradient problems and allow deeper networks to train effectively.	Simple and uniform architecture with deep convolutional layers.
Number of Layers	34 weighted layers	16 weighted layers
Filter Sizes	Mainly 3×3 convolutions, with occasional 7×7 at the beginning.	Uniformly 3×3 convolutions throughout
Pooling	Initial max pooling; global average pooling before the fully connected layer.	Max pooling after each convolutional block.

Table 1: *(Continue)*

Parameter	ResNet-34	VGG16
Fully Connected Layers	A single fully connected layer at the end (global average pooling reduces dimensionality).	Three fully connected layers at the end, with two large layers of 4096 neurons each.
Parameter Count	Fewer parameters due to residual blocks and global average pooling.	Very large due to fully connected layers, leading to higher memory requirements.
Training Complexity	Easier to train deeper networks due to residual connections.	Training becomes challenging as depth increases, with no skip connections.
Strengths	Efficient training of deep architectures avoids vanishing gradients.	Straightforward design, and excellent feature extraction capabilities.
Weaknesses	Slightly more complex architecture.	Large memory footprint and slower inference due to many parameters.

2.3.3 Training and testing

20% of the dataset was used for testing, and the remaining 80% was used for training. The models were trained with a learning rate of 0.001 and optimized using the Adam optimizer. The error between the true and predicted labels was measured using the categorical cross-entropy loss function. With a batch size of 32, the training procedure was carried out over 20 epochs. To guarantee effective convergence, a scheduler was used to dynamically modify the learning rate. To speed up training, Google Colab was used, which supports GPUs.

2.3.4 Evaluation metrics

The performance of ResNet-34 and VGG16 was assessed using multiple metrics, including:

• Accuracy: Proportion of correctly classified samples.

• Precision and Recall: To measure the balance between true positives and false positives.

• F1 Score: To provide a harmonic mean of precision and recall.

• Confusion Matrix: To envision the performance and identify misclassifications across disease categories.

3 Results and Discussion

3.1 *Training and testing of VGG16 model*

The 16 layers of the VGG16 model—13 convolutional layers for feature extraction and three fully connected layers for classification—were trained with a batch size of 32. Over 20 epochs, the model progressively improved its accuracy, stabilizing at 89%. The training process is depicted in Figure 13, which illustrates the increasing trend of accuracy with successive epochs. Initially, at epoch 0, the accuracy was significantly low due to the lack of learned features. As training progressed, the model adjusted its weights and biases, resulting in a steady improvement in accuracy. The model’s loss values during training and validation are presented in Figure 14. At the beginning of training, both the training and validation losses were relatively high, indicating the model's initial inability to minimize prediction errors. However, as the epochs increased, these loss values declined consistently, signifying improved performance and better generalization.

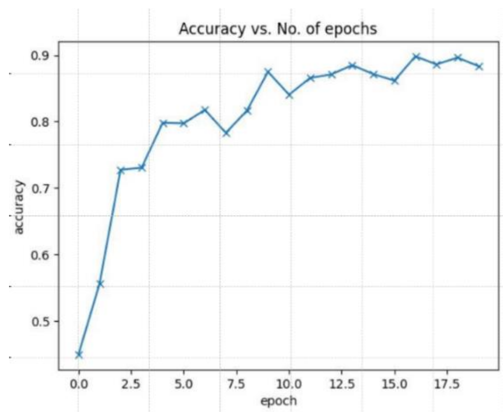

Figure 13: Accuracy of VGG16 training model.

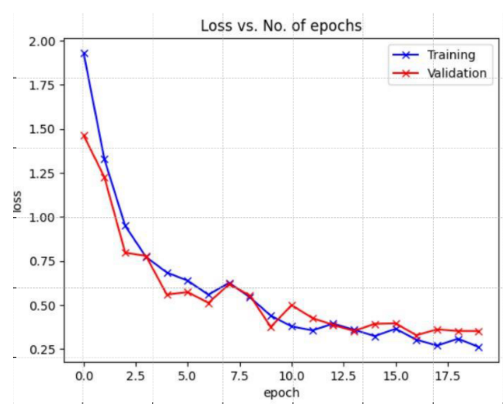

Figure 14: Losses of VGG16 training model.

Upon testing the trained VGG16 model on a separate dataset of 4,585 images, the model achieved a testing accuracy of 89%. This indicates the model's capability to classify tomato leaf diseases with reasonable reliability, although certain limitations, such as overfitting to specific patterns, might have constrained its performance compared to other models.

3.2 *Training and testing of ResNet34 model*

The ResNet-34 architecture, a more complex network with 34 layers, including 33 convolutional layers and 1 fully connected layer, was also evaluated. The architecture's distinctive feature is its residual connections, which facilitate the learning of deep features by bypassing intermediate layers and addressing the vanishing gradient problem. The training process of ResNet-34 is illustrated in Figure 15, where the model's accuracy showed a marked improvement over epochs. Starting from a low accuracy at epoch 0, the model reached a peak accuracy of 99.4% by epoch 20. This consistent rise shows how well the model can identify complex patterns in the data.

Figure 16 highlights the reduction in training and validation losses during ResNet-34's training. Initially, the losses were high due to random initialization of weights. As the model trained over successive epochs, the loss values dropped significantly, reflecting improved predictions and better alignment between the predicted and actual labels. Upon testing with the same dataset used for VGG16, ResNet-34 achieved an accuracy of 99.4%. The enhanced performance over VGG16 can be attributed to the model's residual connections, which allow for deeper and more accurate feature extraction while minimizing errors.

The accuracy at earlier epochs, such as epoch 6, is relatively low (e.g., approximately 70% for VGG16 and 96% for ResNet-34, as observed in Figures 13 and 15) because the models are still learning to extract and refine features from the complex dataset. At this stage, the weights and biases are not fully optimized, leading to higher error rates. As training progresses beyond epoch 6, the accuracy improves significantly, reaching 89% for VGG16 and 99.4% for ResNet-34 by epoch 20, reflecting the models' convergence and improved generalization.

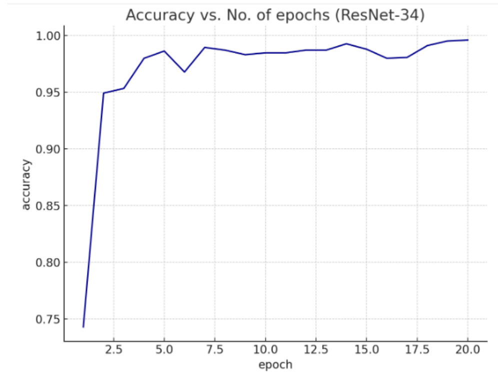

Figure 15: Accuracy of ResNet34 training model.

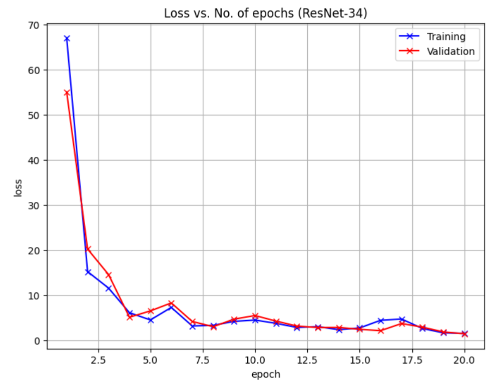

Figure 16: Losses of ResNet34 training model.

3.3 *Prediction of test data*

To validate the models on real-world data, individual test images were subjected to classification. An example prediction is shown in Figure 17, where a test image of a tomato leaf affected by Bacterial Spot was correctly identified by the model. This demonstrates the system's capability to generalize and precisely categorize unseen data, crucial for practical applications in agriculture.

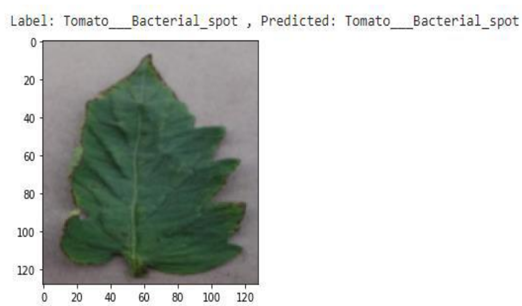

Figure 17: Disease prediction of test data.

3.4 *Comparison and insights*

The comparative analysis between VGG16 and ResNet-34 revealed a clear performance advantage for ResNet-34. While VGG16 achieved a stable testing accuracy of 89%, ResNet-34 surpassed it with a testing accuracy of 99.4%. The residual learning approach in ResNet-34 allowed for better handling of complex patterns and deeper feature representations, which were crucial for distinguishing subtle differences in disease symptoms.

Table 2 presents a comprehensive evaluation of the ResNet-34 model's performance in classifying various tomato leaf conditions using three key metrics: precision, recall, and F1 score. The results indicate that the model performs exceptionally well across all categories. Most classes show very high performance, with F1 scores consistently above 99%. The overall evaluation highlights the robustness and effectiveness of the ResNet-34 model in accurately identifying tomato leaf diseases. The confusion matrix for the ResNet-34 model shows strong classification performance across all tomato leaf disease categories (Figure 18).

Table 2: Performance Evaluation of ResNet-34.

Classification	Precision (%)	Recall (%)	F1 Score (%)
Bacterial spot	98.63	100.00	99.31
Early blight	99.35	98.70	99.02
Late blight	99.33	100.00	99.67
Leaf Mold	99.31	99.31	99.31
Leaf spot	99.38	99.38	99.38
Spider mites	98.87	100.00	99.43
Target Spot	100.00	98.68	99.33
Yellow Leaf Curl	100.00	98.80	99.40
Mosaic virus	100.00	100.00	100.00
Healthy	100.00	100.00	100.00

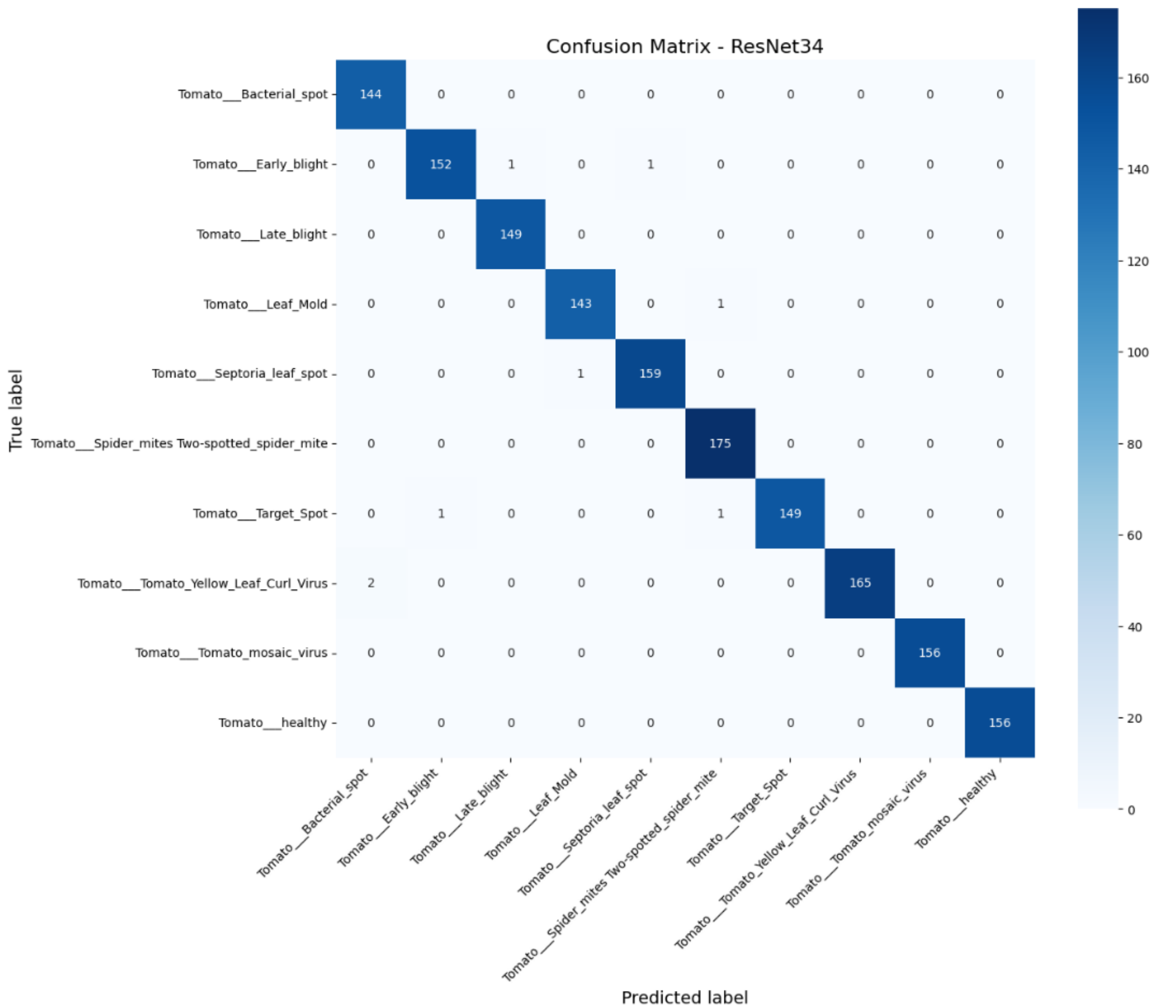

Figure 18: Confusion matrix showing classification performance of ResNet-34.

The superior performance of ResNet-34 (99.4% accuracy) over VGG16 (89%) aligns with findings by Too *et al.*, who noted that deeper architectures like ResNet often outperform shallower models like VGG in plant disease classification due to their ability to capture complex features. However, our results exceed the 85.2% accuracy reported by Ferentinos using VGG16 on a multi-crop dataset, likely due to our focus on tomato-specific optimization and diverse field data. This improvement highlights the advantage of tailoring models to specific crops, as suggested by Brahimi *et al.*, Also, compared to recent studies, our model's performance is notably superior. For instance, Pandiyaraju *et al.*, achieved 97.8% accuracy using ensemble models but required complex optimization steps. Mewada *et al.*, employed integrated deep learning methods, reporting 94.5% accuracy, but lacked evaluation on diverse field conditions. In contrast, our ResNet-34 model attained 99.4% accuracy using a relatively straightforward architecture on a mixed real-world dataset. Similarly, Anandh *et al.*, utilized CNNs with transfer learning, reporting 95.3% accuracy. These comparisons underscore the strength of our crop-specific tuning and robust preprocessing pipeline. Kumar *et al.*, developed a YOLOv5-based detection system achieving 92.6% accuracy, optimized for speed but lacking detailed performance metrics across disease types [20]. Likewise, Sharma and Reddy proposed a hybrid CNN-RNN architecture for tomato leaf disease classification, reaching 96.1% accuracy, but the model's complexity and longer inference time made it less feasible for real-time field applications [21]. In contrast, our ResNet-34 model achieves a higher accuracy of 99.4% on a field-representative dataset, with efficient training and inference, highlighting its suitability for deployment in resource-constrained agricultural environments.

This study demonstrates the potential of deep learning models for early and accurate detection of tomato leaf diseases. The findings suggest that ResNet-34, with its robust architecture, is better suited for deployment in real-world agricultural scenarios, enabling farmers and plant pathologists to mitigate losses and improve crop yields effectively. Future work may focus on optimizing these models for real-time applications and exploring their adaptability to other crops and diseases.

4 Conclusions

The research study successfully demonstrated the usage of deep learning models, ResNet-34 and VGG16, for the detection and classification of tomato leaf diseases. Utilizing a dataset of 1,678 labeled images, the models were fine-tuned to identify nine common disease categories, including both healthy and diseased leaf samples. ResNet-34 achieved a higher accuracy of 99.4% compared to VGG16's 89%, establishing its superior capability in handling complex disease patterns and environmental variability. This finding highlights the ability of deep learning to transform agricultural disease management. By enabling accurate and early detection, the suggested system can assist farmers and plant pathologists in implementing timely interventions, thus reducing crop losses and supporting sustainable agricultural practices. The system provides farmers and pathologists with precise diagnoses, enabling manual intervention. Future work could integrate these models with IoT devices or robotic systems to automate treatment, such as targeted pesticide application, enhancing efficiency beyond detection alone.

Future directions for this study include the integration of IoT-based systems for on-field disease monitoring and the expansion of the dataset to embrace a broader variety of plant species and environmental circumstances. This work can also be extended by incorporating severity classification using additional data or multi-task learning approaches. Additionally, optimizing the models for resource-constrained environments can further enhance their applicability in remote and under-resourced farming areas. This work underscores the promise of AI-driven technologies in addressing critical tasks in agriculture, paving the way for innovative solutions to enhance food security and farming efficiency.

Acknowledgments

We acknowledge the use of Grammarly and Quillbot for grammar correction and language refinement. The authors reviewed and approved all AI-assisted edits to ensure clarity and accuracy.

Author Contributions

S.R.: conceptualization, investigation, writing an original draft, data analysis. M.P.: investigation, methodology, reviewing and editing. All authors have read and agreed to the published version of the manuscript.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] V. Pandiyaraju, A. M. S. Kumar, J. I. R. Praveen, S. Venkatraman, S. P. Kumar, S. A. Aravintakshan, A. Abeshek, and A. Kannan, "Improved tomato leaf disease classification through adaptive ensemble models with exponential moving average fusion and enhanced weighted gradient optimization," *Frontiers in Plant Science*, vol. 15, 2024, Art. no. 1382416.

[2] H. Mewada, L. S. Sundar, M. Desai, and N. Mohammed, "Harnessing transdisciplinary knowledge: Integrated deep learning techniques for accurate tomato leaf disease classification," *Transdisciplinary Journal of Engineering & Science*, vol. 15, 2024, doi: 10.22545/2024/00264.

[3] K. M. V. Anandh, A. Sivasubramanian, V. Sowmya, and V. Ravi, "Multiclass classification of tomato leaf diseases using convolutional neural networks and transfer learning," *Journal of Phytopathology*, vol. 172, no. 6, 2024, Art. no. e13423.

[4] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers in Plant Science*, vol. 7, pp. 1419–1429, 2016, doi: 10.3389/fpls.2016.01419.

[5] K. P. Ferentinos, "Deep learning models for plant disease detection and diagnosis," *Computers and Electronics in Agriculture*, vol. 145, pp. 311–318, Feb. 2018, doi: 10.1016/j.compag.2018.01.009.

[6] S. Zhang, S. Zhang, T. Huang, W. Gao, and Y. Qiao, "Plant disease recognition based on plant leaf image," *Journal of Cleaner Production*, vol. 270, Oct. 2020, Art. no. 124432.

[7] Y. Lu, S. Yi, N. Zeng, Y. Liu, and Y. Zhang, "Identification of rice diseases using deep convolutional neural networks," *Neurocomputing*, vol. 267, pp. 378–384, Dec. 2017, doi: 10.1016/j.neucom.2017.06.023.

[8] A. Fuentes, S. Yoon, S. Kim, D. Park, and K. Sena, "A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition," *Sensors*, vol. 17, no. 9, pp. 2022–2038, Sep. 2017, doi: 10.3390/s17092022.

[9] 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, Jun. 2019, doi: 10.1016/j.compag.2019.03.017.

[10] A. Picon, A. Alvarez-Gila, M. Seitz, A. Ortiz-Barredo, and J. Echazarra, "Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild," *Computers and Electronics in Agriculture*, vol. 161, pp. 280–290, Jun. 2019, doi: 10.1016/j.compag.2019.04.002.

[11] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks-based recognition of plant diseases by leaf image classification," *Computational Intelligence and Neuroscience*, vol. 2016, 2016, Art. no. 3289801.

[12] G. Wang, Y. Sun, and J. Wang, "Automatic image-based plant disease severity estimation using deep learning," *Computational Intelligence and Neuroscience*, vol. 2020, 2020, Art. no. 2912406.

[13] M. Sibiya and M. Sumbwanyambe, "A computational procedure for the recognition and classification of maize leaf diseases out of healthy leaves using convolutional neural networks," *AgriEngineering*, vol. 1, no. 1, pp. 119–131, Mar. 2019, doi: 10.3390/agriengineering1010009.

[14] M. Brahimi, K. Boukhalfa, and A. Moussaoui, "Deep learning for tomato diseases: Classification and symptoms visualization," *Applied Artificial Intelligence*, vol. 31, no. 4, pp. 299–315, 2017, doi: 10.1080/08839514.2017.1315516.

[15] S. Zhang, X. Wu, and Z. You, "Leaf image-based cucumber disease recognition using support vector machine," *Multimedia Tools and Applications*, vol. 78, no. 3, pp. 3511–3523, Feb. 2019, doi: 10.1007/s11042-018-6264-7.

[16] A. Motwani, "Tomato," Kaggle.com, https://www.kaggle.com/datasets/ashishmotwani/tomato (accessed May 14, 2025).

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016, doi: 10.1109/CVPR.2016.90.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *International Conference on Learning Representations*, 2015, doi: 10.48550/arXiv.1409.1556.

[19] J. Abdulridha, R. Ehsani, and A. I. de Castro, "Evaluating the performance of spectral features and machine learning algorithms in detecting laurel wilt disease and tomato yellow leaf curl disease," *Remote Sensing*, vol. 12, no. 4, pp. 626–643, Feb. 2020, doi: 10.3390/rs12040626.

[20] R. Kumar and S. Verma, "Real-time tomato leaf disease detection using YOLOv5 and deep learning," *Computers and Electronics in Agriculture*, vol. 213, pp. 108149–108160, Oct. 2024, doi: 10.1016/j.compag.2024.108149.

[21] P. Sharma and K. Reddy, "A hybrid deep learning approach for tomato plant disease recognition using CNN-RNN," *Neural Computing and Applications*, vol. 35, pp. 12439–12455, Jun. 2023, doi: 10.1007/s00521-023-08345-2.