Fine-Grained Tomato Leaf Disease Classification Using Swin Transformer V2 on the PlantVillage Dataset

Mr. Gopal Bhojak[1], Dr. Ruchi Dave[2], Dr. Pratibha Soni[3]

[1]M.Tech. CSE Scholar, Department of Computer Science, Jagan Nath University, Jaipur
[2]Assistant Professor-CSE, Department of Computer Science, Jagan Nath University, Jaipur
[3]Assistant Professor-CSE, Department of Computer Science, Jagan Nath University, Jaipur

Abstract:
Tomato leaf diseases remain a major cause of yield loss in one of the most widely cultivated vegetable crops, and manual, expert-dependent diagnosis is slow, inconsistent, and often inaccessible in resource-limited farming regions. This study proposes a fine-tuned Swin Transformer V2 model for automated tomato leaf disease classification on the existing PlantVillage dataset, retaining the full nine-disease-plus-healthy (ten-class) taxonomy rather than the reduced class groupings adopted by several comparable studies. The model was trained using an ImageNet-pretrained swinv2-tiny backbone, cross-entropy loss with label smoothing. On a test split of 2,403 images, the model achieved 99.71% accuracy, with weighted precision, recall, and F1-score each equal to 99.71%, misclassifying only six images — predominantly between visually similar disease pairs. Compared against two related studies using the same or a related PlantVillage tomato taxonomy, the proposed model achieved a higher clean-test accuracy than both. These results indicate that hierarchical vision-transformer architectures are a competitive approach for fine-grained tomato leaf disease classification and merit further investigation for practical agricultural deployment.

Keywords: Tomato Leaf Disease; Deep Learning; Swin Transformer V2; Vision Transformer; PlantVillage Dataset; Precision Agriculture.

1. Introduction

In Agriculture the crop productivity is persistently undermined by plant diseases, a major contributor to yield loss worldwide. Tomato is among the most widely cultivated crops, is particularly vulnerable to bacterial, fungal, oomycete, and viral leaf diseases, several of which produce visually similar early-stage symptoms difficult to distinguish through manual inspection.

Manual disease diagnosis relies on visual assessment by agricultural experts, and it is time-consuming and impractical at the scale of modern farming, particularly where plant-pathology expertise is limited. Deep learning enables automated, image-based disease classification without hand-engineered features [1]. Convolutional neural networks (CNNs) such as VGG, ResNet, Inception, DenseNet, and MobileNet have been widely applied to this problem, frequently achieving accuracies above 95% on the PlantVillage benchmark dataset [1], [2]. Despite their effectiveness in image classification, CNN-based models primarily focus on localized features, which may limit their ability to establish relationships between distant regions of an image when disease symptoms are distributed across the leaf surface

Vision Transformers (ViTs) [4] address this limitation via self-attention across image patches, but at a computational cost that limits practicality on smaller agricultural datasets. The Swin Transformer [4]and its successor Swin Transformer V2 [5] mitigate this cost through hierarchical, shifted-window self-attention, combining CNN-like multi-scale representation with transformer-style long-range modelling. Standalone Swin Transformer backbones nonetheless remain comparatively underexplored in the tomato leaf disease literature relative to CNN-based approaches [8].

This study addresses that gap by fine-tuning a Swin Transformer V2 backbone on the full ten-class PlantVillage tomato taxonomy, rather than a simplified class grouping as adopted in related work [1], [2], to evaluate performance on a more realistic classification problem. Section 2 reviews related literature work; Section 3 describes the materials and methodology; Section 4 discusses results; Section 5 conclude the work.

2. Related Work

A substantial body of work has applied CNN transfer learning for tomato's leaf disease classification on the Plant-Village dataset. Krishna Kumar et al. [2] fine-tuned a VGG19 architecture, reporting 91.56% validation and 98.36% test accuracy. Sharma et al. [3] used an ensemble combining ResNet50 and MobileNetV2, reporting 99.91% test accuracy on a ten-class, 11,000-image tomato dataset, though an internal inconsistency between two of the study's own results tables was noted. Begum et al. [4] proposed a hybrid ResNet50–PCA–SVM pipeline on the full 38-class PlantVillage dataset, reporting 89.4% validation accuracy and 98.63% mean accuracy under five-fold cross-validation. Numerous further CNN-based studies [9]–[20] report accuracies generally between 91% and 98% using architectures including InceptionV4, EfficientNet, MobileNet, LeNet, and custom CNN designs, typically evaluated on clean PlantVillage imagery.

A second line of work incorporates attention mechanisms into CNN backbones to improve localisation of disease-relevant regions. Dhanya et al. [6] proposed a spatial-attention-augmented CNN, though the study's own results table (91%) diverges from its abstract's headline claim (99%). Zhang et al. [21], Li et al. [22], Zhao et al. [23], Wang et al. [24], Chen et al. [25], and Singh et al. [26] report attention-augmented architectures achieving 95.6–98.5% accuracy, generally two to four percentage points above non-attention CNN baselines, with Singh et al. [26] additionally incorporating Grad-CAM-based explainability.

Transformer-based approaches remain comparatively rare in this domain. Ghosh et al. [6] proposed a hybrid Vision Transformer/EfficientNetV2/Swin Transformer architecture reporting 98% accuracy on a custom, non-public, grayscale-preprocessed dataset. TabiaTanzinPrama and Oni [7] contribute one of the few field-acquired tomato leaf datasets identified in the literature, though framed as a binary (healthy/diseased) task on a small (1,028-image) sample. Two recurring characteristics motivate the present study: reduced class granularity in several works [1], [5], which limits comparability with the full nine-disease taxonomy; and the comparative scarcity of standalone vision-transformer backbones evaluated on this problem [7], addressed directly here.

3. Materials and Methods

3.1 Methodology

The presented methodology in Fig. 1, begins with the Tomato Leaf Dataset, containing 16,011 images distributed across 10 classes. The dataset splited into 70%- training, 15%- validation, and 15% -testing subsets. The dataset then preprocessed by resizing images to pixel (224 × 224), applied normalization, and data augmentation only on the training set. Subsequently, a Swin Transformer V2 model is fine-tuned using the AdamW -optimizer, loss function, and cosine learning-rate scheduler. The best-performed model is selected based on validation accuracy, and final performance evaluated on test set using performance matrices.

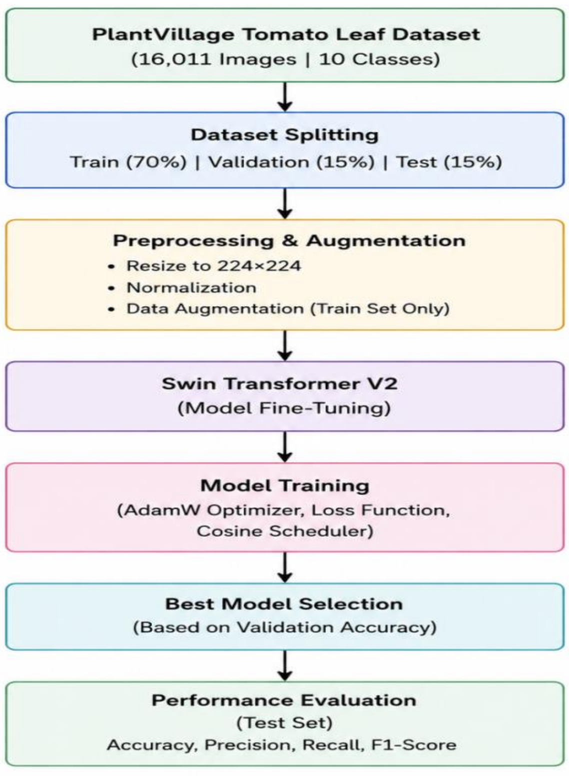

Fig.1 *Proposed Methodology*

3.2 Dataset

The PlantVillage image repository , accessed via its Kaggle mirror, was used as the data source. This study is restricted to the tomato subset, comprising 10 classes , as detailed in Table 1.

Table 1. Tomato disease class taxonomy and image counts.

S.No.	Disease Class	No. of Images	Category
1	Tomato_Bacterial_spot	2127	Bacterial
2	Tomato_Early_blight	1000	Fungal
3	Tomato_Late_blight	1909	Oomycete
4	Tomato_Leaf_Mold	952	Fungal
5	Tomato_Septoria_leaf_spot	1771	Fungal
6	Tomato_Spider_mites	1676	Mite
7	Tomato__Target_Spot	1404	Fungal
8	Tomato__Tomato_YellowLeaf__Curl_Virus	3209	Viral
9	Tomato__Tomato_mosaic_virus	373	Viral
10	Tomato_healthy	1591	—
	TOTAL SAMPLES	**16012**	

3.3 Preprocessing and Augmentation

Dataset images were resized to px 224 × 224 — rather than the backbone's native 256 × 256 — and normalised using the pretrained Swin V2 processor's statistics. This resolution was chosen because the Swin V2-tiny backbone [4], [5]partitions inputs into 32-px windows (patch 4 × window 8); since 224 = 7 × 32, the window grid still divides evenly without padding artefacts. Training images were augmented with random-resized cropping, horizontal flipping, rotation, colour jitter, and random autocontrast, equalisation, and sharpness adjustment. Validation and test images received deterministic resizing and normalisation only, ensuring metrics reported across training epochs remain directly comparable.

3.4 Model Architecture

The classification backbone is Swin Transformer V2 (tiny variant, swinv2-tiny-patch4-window8-256) [4], [5], a vision transformer that computes self-attention within shifted local windows, achieving linear rather than quadratic computational scaling with image size while retaining long-range dependency modelling through window shifting across layers. The model was initialised with ImageNet-1k pretrained weights, and its original head was replaced by a newly initialized 10-class linear layer. All backbone parameters were fine-tuned end-to-end (not frozen).

3.5 Training Configuration

The complete training configuration is summarised in Table 2. The model was fine-tuned for 10 no. of epochs using cross-entropy loss with label smoothing ($\alpha = 0.1$) [27] and the AdamW-optimizer [28], with a cosine-decay learning-rate schedule preceded by a linear warm-up over the first 10% steps of training. The checkpoint achieving the highest validation accuracy was retained for final test evaluation.

Table 2. Training hyperparameter configuration.

Hyperparameter	**Value**
Backbone	swinv2-tiny-patch4-window8-256 (ImageNet-1k pretrained)
Input resolution	224 × 224 px
Output classes	10
Loss- function	Cross-entropy with label smoothing ($\alpha = 0.1$) [35]
Optimizer	AdamW- (weight decay :0.01) [34]
Learning -rate	2×10^{-5}
LR schedule	Cosine decay with linear warm-up (10% of steps)
Batch size / Epochs	32 / 10

3.6 Evaluation Protocol

Model was evaluated using accuracy, precision, recall, and F1-score (weighted averages across the ten classes), a full per-class classification report, and a confusion matrix on the test split. This combination of aggregate and per-class reporting follows recommended practice for multi-class classification evaluation, providing a more complete assessment than accuracy alone.

4. Results and Discussion

4.1 Training and Validation Performance

Training and validation loss decreased sharply within the first two epochs and remained closely aligned throughout training, indicating minimal overfitting as presented in Fig. 1.2. Validation accuracy rose rapidly during the first two to three epochs before plateauing between 0.996 and 0.997 for the remainder of training, with the highest validation accuracy (0.9971) recorded at epoch 8; this checkpoint was retained for final evaluation. The early plateau suggests that the ten-epoch training budget exceeded what was strictly necessary for convergence on this dataset.

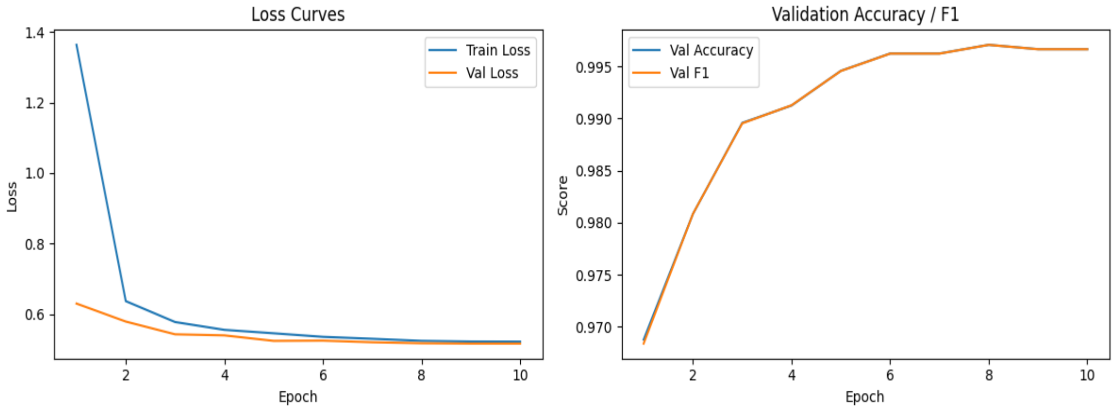

Fig 2. *Training and validation performance during model's training*

4.2 Test-Set and Per-Class Performance

On the held-out test split, the best checkpoint achieved 99.71% accuracy, with weighted precision, recall, and F1-score each equal to 99.71%, consistent with its validation-time performance and indicating reliable generalisation rather than an optimistic validation estimate. Table 3 reports the complete per-class breakdown. Four of the ten classes —" late blight, leaf mold, mosaic virus, and healthy" — achieved perfect precision and recall; the remaining six classes scored 0.99 or 1.00 on both metrics. The mosaic virus class, despite a perfect score, has the smallest test-set support (46 images versus 498 for yellow leaf curl virus), and its reported metrics should be interpreted with corresponding caution given the small sample size.

Table 3. Per-class precision, recall, F1-score, and support on the held-out test split.

Class	**Precision**	**Recall**	**F1-Score**	**Support**
Bacterial Spot	0.99	1	0.99	315
Early Blight	1	0.99	1	170
Late Blight	1	1	1	271
Leaf Mold	1	1	1	120
Septoria Leaf Spot	1	1	1	265
Spider Mites	1	1	1	258
Target Spot	0.99	1	1	205
Yellow Leaf Curl Virus	1	0.99	1	498
Mosaic Virus	1	1	1	46
Healthy	1	1	1	255
Weighted Avg.	1	1	1	2,403

4.3 Confusion Matrix Analysis

Fig. 3 represents that only six of 2,403 test images were misclassified (≈ 0.25% error rate), concentrated in two patterns: four images were confused with bacterial spot (one from early blight, three from yellow leaf curl virus), and two images were confused with target spot (one from septoria leaf spot, one from spider mites). Both confusion patterns occur between disease categories that share overlapping visual symptoms (small dark lesions and concentric necrotic ring patterns, respectively), consistent with the general pattern reported across the literature [1], [3] that residual errors in high-performing classifiers concentrate among visually similar disease pairs rather than being distributed randomly.

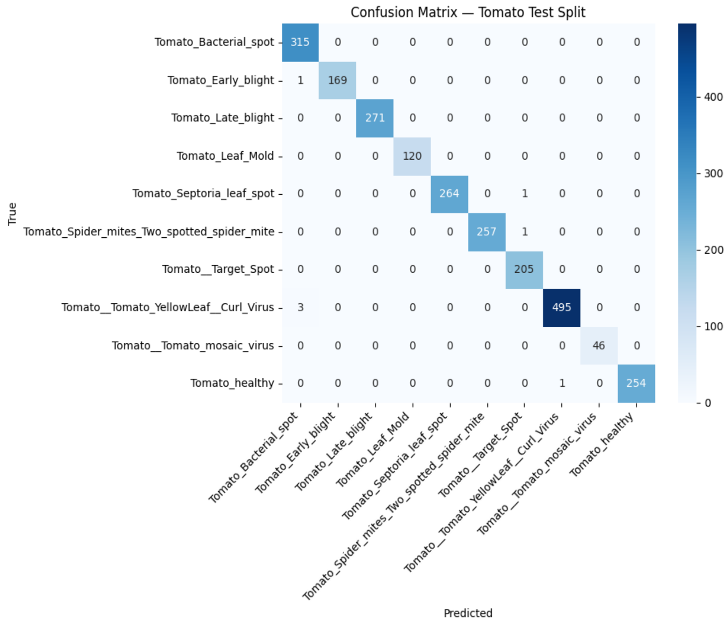

Fig. 3 *Confusion matrix*

4.4 Comparison with Existing Studies

Table 4 compares the present study's results against related works from the literature reviewed in Section 2. Mohebbanaaz et al. [1] evaluate a six-class grouped taxonomy that is inherently easier than the ten-class taxonomy retained here, so their lower reported accuracy (97.00%/98.17%) partly reflects reduced task difficulty rather than model capability alone. Krishna Kumar et al. [2] use the same ten-class taxonomy, making their 98.36% test accuracy the more directly comparable figure; the present study's 99.71% represents a modest improvement.

*Table 4. Comparison of reported test accuracy with related studies. *Denotes a figure drawn from a study with a noted internal inconsistency (see Section 2).*

Study	Architecture	Classes	Test Accuracy (%)
Present study	Swin Transformer V2	10	99.71
Mohebbanaaz et al. (2025) [1]	CNN (InceptionV3 / ResNet56)	6 (grouped)	97/ 98.17
Krishna Kumar et al. (2026) [2]	CNN (VGG19)	10	98.36

5. Conclusion and Future Work

This work fine-tuned a Swin Transformer V2 model for fine-grained, ten-class tomato leaf disease classification on kaggel dataset: PlantVillage dataset, achieving 99.71% test accuracy with only six misclassifications out of 2,403 test images, outperforming two related studies using comparable or identical class taxonomies. These results support hierarchical vision-transformer architectures as a competitive approach for fine-grained agricultural image classification, retaining the full disease taxonomy rather than a simplified grouping.

Future work should extend this evaluation to field-acquired imagery to assess performance under variable lighting, backgrounds, and camera conditions, and should assess model robustness under simulated or real distributional shift (e.g., blur, noise, adverse weather), which remains largely unexplored in this literature. Further directions include benchmarking additional transformer architectures (e.g., MaxViT, EVA-02, AIMv2, DINOv3) under an identical protocol, extending the approach to other crops, and deploying the trained model within a farmer-facing mobile or web application.

REFERENCES:

[1] Mohebbanaaz, K. Mounika, N. Sadaf, and V. Preethi, "Tomato Leaf Disease Prediction and Classification using Deep Neural Network (DNN)," 2025 6th International Conference on Recent Advances in Information Technology (RAIT), 2025.

[2] Krishna Kumar, M. V. Gupta, A. Kumar, A. Bishnoi, and M. Joshi, "Deep Learning Based Tomato Leaves Disease Detection: A Comprehensive Approach Using Convolutional Neural Networks," Journal of Recent Innovations in Computer Science and Technology, vol. 3, no. 2, pp. 22-30, 2026.

[3] J. Sharma et al., "Deep learning based ensemble model for accurate tomato leaf disease classification by leveraging ResNet50 and MobileNetV2 architectures," Scientific Reports, vol. 15, p. 13904, 2025.

[4]Z. Liu et al., "Swin Transformer: Hierarchical Vision Transformer using Shifted Windows," Proc. IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 10012-10022.

[5]Z. Liu et al., "Swin Transformer V2: Scaling Up Capacity and Resolution," Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 12009-12019.

[6] R. Dhanya, S. Mythili, X. Rexeena, and D. Devadas, "Detection of Tomato Leaf Diseases Using Deep Learning and Spatial Attention Mechanisms," International Journal of Research and Scientific Innovation, vol. 12, no. 7, pp. 1144-1153, 2025.

[7] TabiaTanzinPrama and M. K. Oni, "Optimized Custom CNN for Real-Time Tomato Leaf Disease Detection," in Computational Science - ICCS 2025 Workshops, Part III, Springer, pp. 81-88, 2025.

[8] S. C. Karuturi et al., "Tomato Leaf Disease Detection Using Deep Learning," Proceedings of the 3rd International Conference on Futuristic Technology (INCOFT 2025), vol. 3, pp. 675-682, 2025.

[9] B. Sowmya and S. Guruprasad, "Deep learning based plant health disease detection in tomatoes using inception v4 convolutional neural network and YOLO V8," Discover Artificial Intelligence, vol. 5, p. 278, 2025.

[10] T. Vijayakumar, S. Santhi, and M. Gokila, "A Deep Learning-based Approach for Tomato Plant Leaf Disease Detection and Classification," 2022 3rd ICOSEC, pp. 1251-1255, 2022.

[11] A. Abdullah et al., "A Deep-Learning-Based Model for the Detection of Diseased Tomato Leaves," Agronomy, vol. 14, no. 7, p. 1593, 2024.

[12] Z. Ullah et al., "EffiMob-Net: A Deep Learning-Based Hybrid Model for Detection and Identification of Tomato Diseases Using Leaf Images," Agriculture, vol. 13, p. 737, 2023.

[13] A. Guerrero-Ibáñez and A. Reyes-Muñoz, "Monitoring Tomato Leaf Disease through Convolutional Neural Networks," Electronics, vol. 12, p. 229, 2023.

[14] Prajwala T. M. et al., "Tomato leaf disease detection using convolutional neural networks," IC3 2018, pp. 1-5, 2018.
[15] 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.
[16] M. Agarwal et al., "ToLeD: Tomato leaf disease detection using convolution neural network," Procedia Computer Science, vol. 167, pp. 293-301, 2020.
[17] K. Zhang, Q. Wu, A. Liu, and X. Meng, "Can deep learning identify tomato leaf disease?" Advances in Multimedia, vol. 2018, pp. 1-10, 2018.
[18] S. Zhao, Y. Peng, J. Liu, and S. Wu, "Tomato leaf disease diagnosis based on improved convolution neural network by attention module," Agriculture, vol. 11, no. 7, p. 651, 2021.
[19] H. S. Nagamani and H. Sarojadevi, "Tomato leaf disease detection using deep learning techniques," IJACSA, vol. 13, no. 1, 2022.
[20] S. G. Paul et al., "A real-time application-based convolutional neural network approach for tomato leaf disease classification," Array, vol. 19, p. 100313, 2023.
[21] Y. Zhang, Y. Zhu, S. Zhang, G. Li, and X. Zhang, "Attention-Guided Deep Learning Framework for Automated Tomato Leaf Disease Detection," IEEE TIM, vol. 72, pp. 1-13, 2023.
[22] Y. Li, X. Qian, J. Huang, Y. Liu, and J. Wang, "Multi-Scale Attention Network for Tomato Leaf Disease Classification," IEEE Access, vol. 10, pp. 44350-44360, 2022.
[23] C. Zhao, C. Guo, J. Zhang, J. Zhang, and Y. Zhu, "Early Detection of Tomato Leaf Diseases Using Self-Attention-Based CNN," IEEE Access, vol. 9, pp. 48827-48836, 2021.
[24] X. Wang, Y. Shen, J. Dong, J. Jiang, and K. Wang, "Lightweight Attention-Based Model for Real-Time Tomato Disease Detection on Mobile Devices," IEEE Access, vol. 10, pp. 32376-32386, 2022.
[25] J. Chen, S. Jiang, P. Zhu, Y. Wei, and H. Chen, "A Spatial Attention-Guided ResNet for Robust Tomato Leaf Disease Detection," IEEE TIM, vol. 71, pp. 1-11, 2022.
[26] R. Singh, P. Singla, and S. Datta, "Explainable AI Model for Tomato Leaf Disease Detection Using Gradient-Weighted Class Activation Mapping and Spatial Attention," IEEE Access, vol. 11, pp. 26133-26144, 2023.
[27] I. Loshchilov and F. Hutter, "Decoupled Weight Decay Regularization," International Conference on Learning Representations (ICLR), 2019.
[28] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the Inception Architecture for Computer Vision," Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2818-2826.