KEC Conference 2026

ISSN: 3021-940X (Print)

DOI: 10.3126/injet.v3i2.95510

Deep-Learning Based Tomato Leaf Disease Classification using CNN, ConvNeXt, Vision Transformer, and Swin Transformer

Niraj Pandey[1*], Denish Oli[2]

[1]*Department of Computer Engineering, Kantipur Engineering College, Dhapakhel, Lalitpur, Nepal,nirajpandey10521@gmail.com*

[2]*Department of Computer Engineering, Kantipur Engineering College, Dhapakhel, Lalitpur, Nepal, denisoli743@gmail.com*

Abstract

Deep-learning-based tomato leaf disease classification remains challenging because it must address visually similar disease patterns, non-target inputs, and computational constraints. This paper presents a deployment-aware comparative study of four deep learning model families—Convolutional Neural Networks (CNN), ConvNeXt, Vision Transformer (ViT), and Swin Transformer—for tomato leaf disease classification. We used a dataset comprising 31,042 training images and 5,643 validation images and include a not_tomato rejection class to better reflect real-world deployment beyond closed-set classification. We also applied HSV-based leaf-focused background removal and evaluated model accuracy, loss behavior, precision, recall, and F1-score. Results showed that Swin Transformer achieved the best overall performance with 98.69% validation accuracy and superior plant detection capability. ViT demonstrated the most stable generalization behavior, while ConvNeXt remained competitive but computationally expensive. The custom CNN offered high efficiency but lower accuracy. These findings highlight the importance of balancing accuracy, generalization, and deployment constraints in agricultural AI systems.

Keywords: Tomato leaf disease classification, deep learning, convolutional neural networks, ConvNeXt, Vision Transformer, Swin Transformer, deployment-aware AI, plant disease detection, agricultural AI, image classification

1. Introduction

Deep learning has significantly advanced plant disease classification, with CNNs initially dominating because of their strong feature-learning ability and transformer-based models such as ViT and Swin later improving global context modeling. ConvNeXt further showed that modernized convolutional designs can remain highly competitive by combining CNN efficiency with transformer-inspired principles. However, real-world deployment remains challenging because many studies rely on controlled datasets, struggle with visually similar diseases, overlook irrelevant non-target inputs, and rarely evaluate computational efficiency.

Tomato leaf disease classification is a suitable benchmark for addressing these issues because it involves overlapping disease symptoms and realistic non-target cases. Therefore, this study compares four model families—custom CNN, ConvNeXt, ViT, and Swin Transformer—on the same cleaned 11-class tomato leaf dataset with a not_tomato category. Beyond classification accuracy, the study also examines generalization behavior, structured confusion, and computational cost to provide a more deployment-aware evaluation of tomato leaf disease detection systems.

2 Related Works

Early studies on tomato leaf disease classification mainly relied on convolutional neural networks (CNNs) because of their strong ability to learn hierarchical visual features. Mohanty et al. reported nearly 99% test accuracy on the PlantVillage dataset, while Chen et al. showed that transfer learning with pre-trained models such as ResNet50 and VGG16 could further improve performance on smaller tomato leaf datasets, with ResNet50 achieving 96.5% accuracy. However, these CNN-based approaches were mostly evaluated

**Corresponding Author*

on controlled datasets with uniform backgrounds and paid limited attention to deployment-related factors such as computational cost, latency, and robustness to irrelevant inputs.

More recently, transformer-based models have also been introduced for plant disease detection. Vision Transformer (ViT), proposed by Dosovitskiy et al., captures global context through self-attention and has shown strong performance in distinguishing visually similar diseases, although at a higher computational cost. Swin Transformer, introduced by Liu et al., improves efficiency through hierarchical shifted-window attention. At the same time, ConvNeXt, also developed by Liu et al., modernizes convolutional design using transformer-inspired principles to balance accuracy and efficiency. Although these models are promising, most prior studies still focus mainly on classification accuracy rather than deployment-aware evaluation.

To address this gap, the present study compares a custom CNN, ConvNeXt, ViT, and Swin Transformer on a cleaned 11-class tomato leaf dataset that includes a not_tomato category. Beyond classification performance, the study also assesses deployment-oriented metrics such as model size, FLOPs, and inference latency, providing a more practical assessment for real-world tomato leaf disease detection systems.

3 Methodology

3.1 Dataset

The shared benchmark contained 31,042 training images and 5,643 validation images across 11 classes: Tomato_Bacterial_spot,Tomato_Early_blight,Tomato_Late_blight,Tomato_Leaf_Mold,Tomato_Septoria_leaf_spot,Tomato_Spider_mites,Tomato_Target_Spot,Tomato_Yellow_Leaf_Curl_Virus,Tomato_Mosaic_virus, Tomato_healthy, and not_tomato. Each disease class contained approximately 2,500 training images, with a variation of about ±200 samples across classes, along with 500 validation images per class. The not_tomato class included 6042 training images and 643 validation images. The not_tomato class contains unrelated non-tomato images such as cars, bears, fruits, and human images. Image-hash checks reported 29,363 unique training hashes and 5,558 unique validation hashes, with zero detected overlap after cleaning.

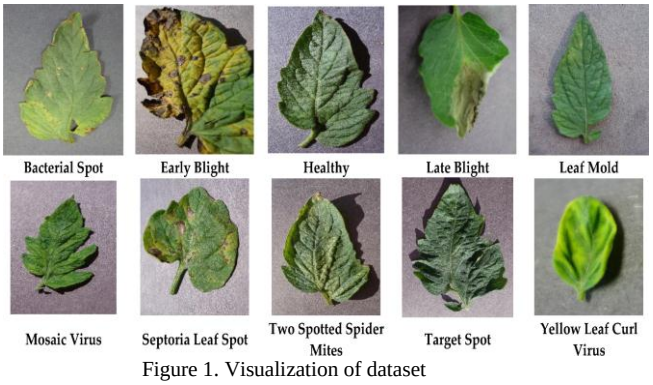

Figure 1. Visualization of dataset

3.2 Preprocessing

All principal models used HSV-based leaf-focused background removal. The pipeline converted RGB to HSV, retained the main green region, applied morphological cleanup, kept the largest connected component, and blacked out the remaining background. This step reduced background distraction and made the benchmark more consistent across model families.

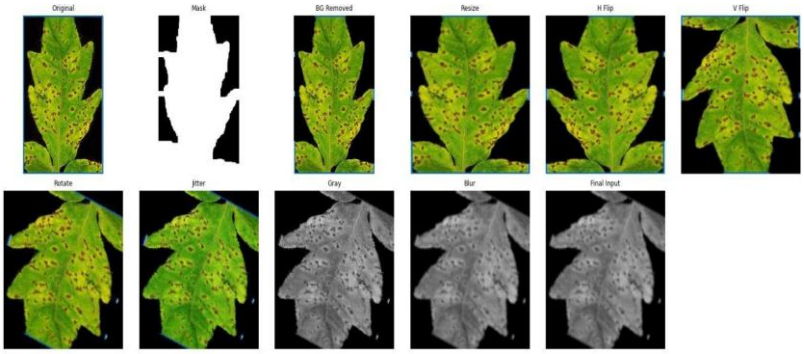

Figure 2. Preprocessed image sample

The validation set was cleaned using perceptual hashing to remove duplicate and near-duplicate overlap with the training set. This step reduced train-validation leakage and made validation performance more reliable.

3.3. Model architecture overview

3.3.1 ViT

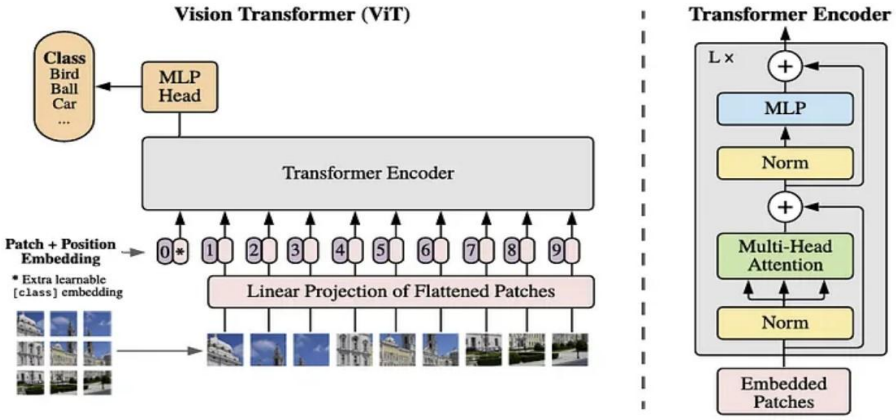

Figure 3. ViT Architecture

The ViT model used in this study is based on a pre-trained Vision Transformer (ViT-Base Patch16-224) architecture. The input image is resized to 224 × 224 and processed by the ViT backbone, which divides the image into fixed-size patches and encodes them through transformer layers. For classification, the CLS token output from the final hidden state is extracted and passed through a dropout layer with a probability of 0.2, followed by a fully connected linear layer to produce predictions for 11 classes. During fine-tuning, the backbone parameters were frozen except for the last six transformer encoder blocks, while the classification head was trained jointly for the final task.

3.3.2 CNN

The CNN model used in this study is a custom architecture composed of three convolutional blocks followed by a classifier. Each convolutional block consists of a 3 × 3 convolution layer, a ReLU activation function, and 2 × 2 max-pooling. The three convolution layers use 32, 64, and 128 filters, respectively. For an input image of size 224 × 224 × 3, the final feature map size becomes 128 × 28 × 28 after the third pooling layer. These feature maps are then flattened and passed through a fully connected layer of 256 neurons, followed by ReLU activation, dropout with a probability of 0.2, and a final linear output layer with 11 neurons for multi-class classification.

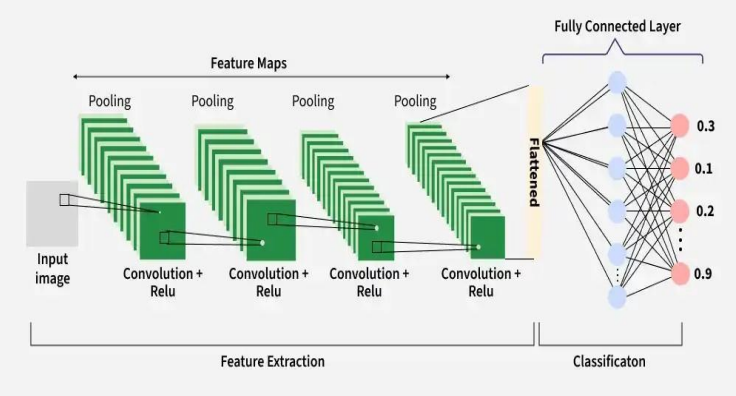

Figure 4. CNN Architecture

3.3.3 ConvNeXt

The ConvNeXt model used in this study is based on a pre-trained ConvNeXt-Base architecture initialized with ImageNet-1K weights. The input images are resized to 160 × 160 and passed to the feature extractor. The backbone consists of stacked ConvNeXt feature blocks with intermediate downsampling stages, followed by the classifier pipeline. For task-specific adaptation, the final classification layer is replaced with a fully connected linear layer for 11 classes. During training, both the backbone and classifier were optimized using separate learning rates.

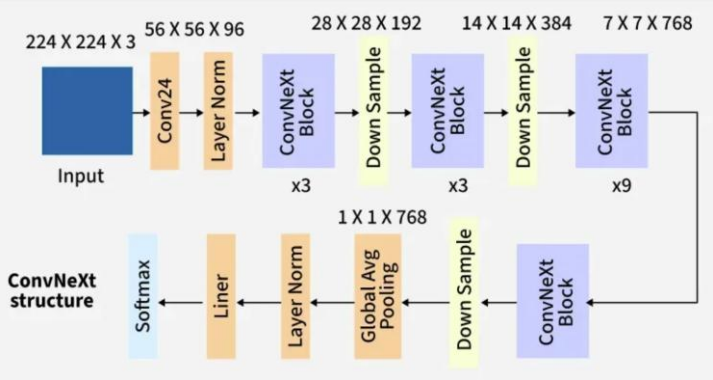

Figure 5. ConvNeXt Architecture

3.3.4 Swim Transformer

The Swin Transformer model used in this study is based on a pre-trained Swin-Tiny Patch4-Window7-224 backbone. The input image is processed by the backbone to obtain a pooled feature representation, which is passed through a custom classification head with a dropout layer (0.2) and a fully connected layer for 11-class prediction. During training, the backbone was initially frozen, with only the last six encoder layers unfrozen for fine-tuning, while the classifier head remained fully trainable.

3.4 Training configuration

All models were trained using a custom LeafImageFolder dataset with HSV-based background removal and model-specific preprocessing transforms. The models were optimized for 20 epochs using the Adam optimizer and cross-entropy loss, and the best-performing weights were selected based on validation accuracy. These configurations were selected after evaluating performance under multiple settings and choosing those that produced the most stable and optimal results for each model. Since the training settings varied across architectures, including input size, batch size, learning rate configuration, fine-tuning strategy, and auxiliary optimization techniques, the detailed configuration for each model is presented in Table 1.

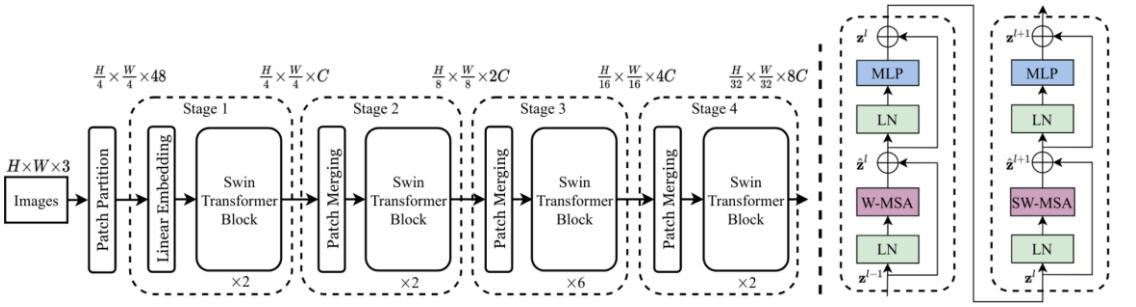

Figure 6. Swim Architecture

Table 1. Training configuration of the evaluated models

Model	Input size	Batch size	Optimizer	Learning Rate	Parameter detail	Epochs
ViT	224×224	32	Adam	1e-5 backbone, 1e-3 for head	last 6 encoder blocks unfrozen	20
CNN	224×224	32	Adam	1e-4 features, 1e-3 classifier	Feature extractor and classifier trainable	20
*-ConvNeXt	160×160	4	Adam	1e-4 features, 1e-3 classifier	Feature extractor and classifier trainable	20
Swin	224×224	32	Adam	1e-5 backbone, 1e-3 head	6 encoder layers unfrozen	20

4 Result

4.1 Generalization Performance

The training and validation curves show clear differences in model behavior. Swin Transformer achieved the best performance, with the highest validation accuracy and lowest loss. ViT showed the most stable learning, with closely aligned training and validation curves and smooth loss reduction. ConvNeXt remained competitive but showed more oscillation in later epochs. In contrast, the custom CNN improved steadily but lagged behind the transfer-learning models in both accuracy and loss, indicating weaker generalization.

4.2 Precision, recall, FI-score and class balance

The quantitative results further confirmed this ranking. Swin achieved the highest weighted precision, recall, and F1-score, followed by ViT and ConvNeXt with similar performance, while CNN had the lowest scores. The close agreement between weighted and macro metrics for Swin, ViT, and ConvNeXt indicates balanced class-wise performance, whereas the larger gap in CNN suggests more uneven classification across classes.

4.3 Confusion matrix interpretation and similar-disease discrimination

The confusion matrices indicate that the mistakes were mostly made in the case of visually similar diseases like Early blight, Septoria leaf spot, Bacterial spot, and Late blight. The symptoms include dark spots, brown lesions, yellow halo, and unevenly affected leaves. Among them, CNN showed the highest confusion among these disease classes, signifying less capability in discerning subtle differences in the disease patterns. ConvNeXt minimized these errors in comparison to CNN, whereas the class separation by ViT was more evenly distributed than other models. Swin Transformer had the most pronounced diagonal confusion pattern among these models, indicating low confusion among visually similar disease classes. It implies that Swin performed well in recognizing details of local lesions and leaf structures. The classes that were relatively easy to classify include those having clear visual traits such as not_tomato, Yellow Leaf Curl Virus, Tomato mosaic virus, Spider mites, and healthy leaves. All in all, Swin performed well as far as confusion matrix is concerned.

4.4 Computational cost and training time

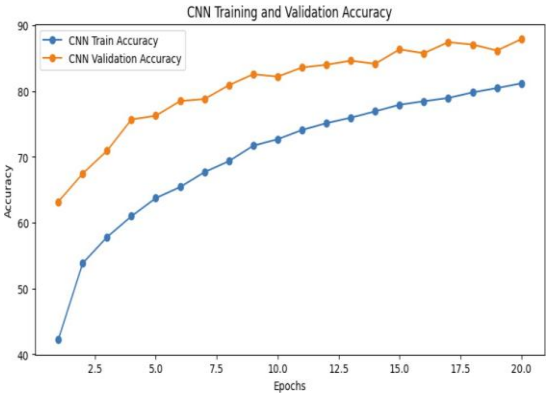

Figure 7. Training and Validation Accuracy for CNN

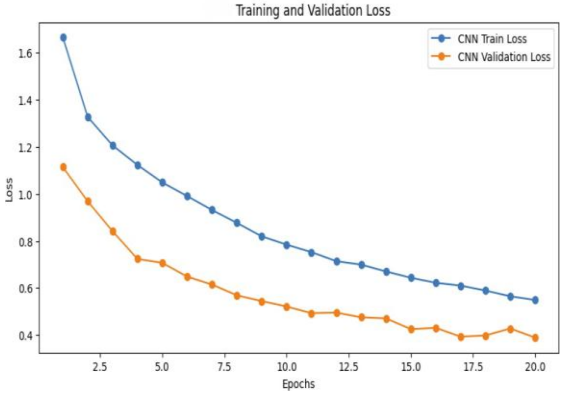

Figure 8. Training and Validation Loss for CNN

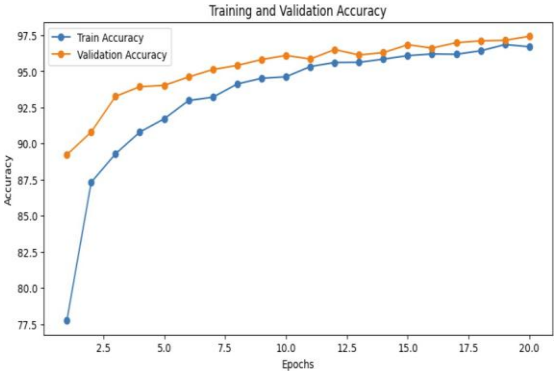

Figure 9. Training and Validation Accuracy for ViT

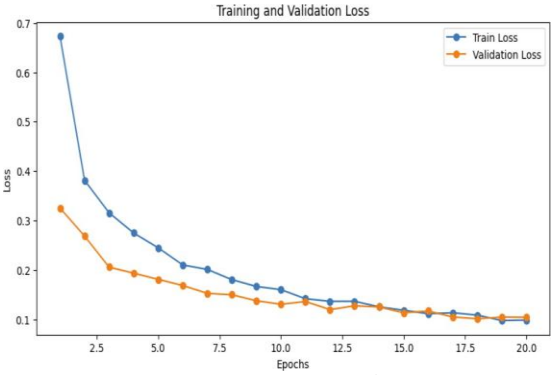

Figure 10. Training and Validation Loss for ViT

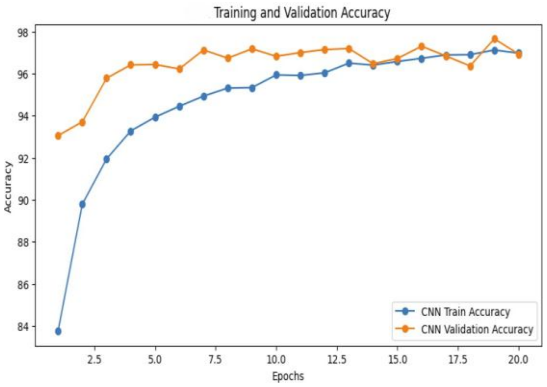

Figure 11. Training and Validation Accuracy for ConvNeXt

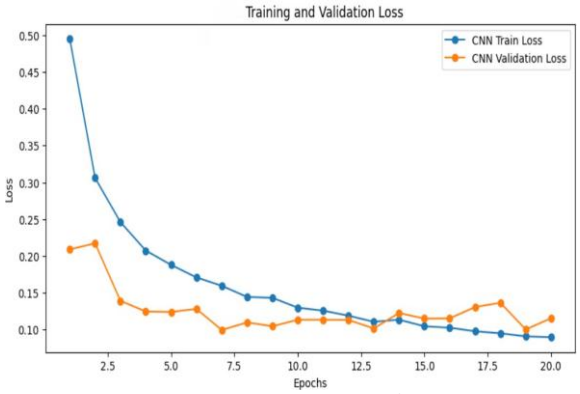

Figure 12. Training and Validation Loss for ConvNeXt

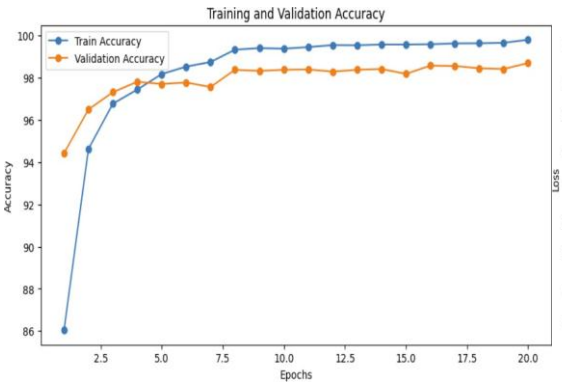

Figure 13. Training and Validation Accuracy for Swin

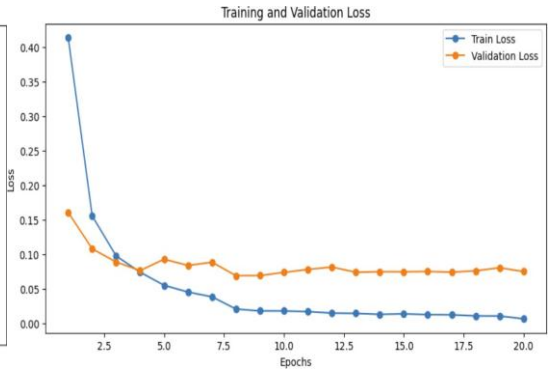

Figure 14. Training and Validation Loss for Swin

4.5 Confusion Matrix

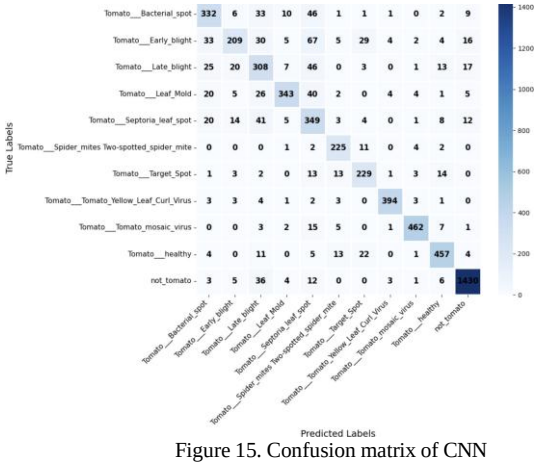

Figure 15. Confusion matrix of CNN

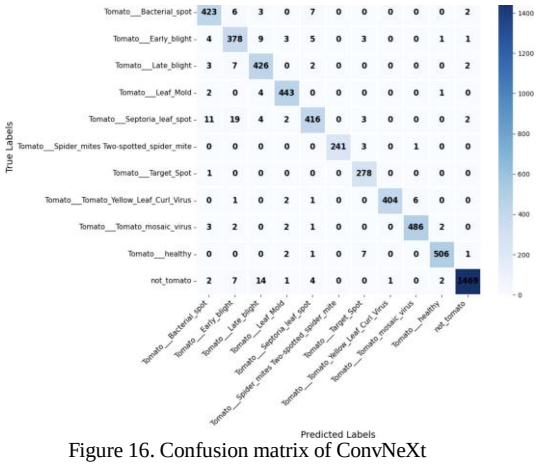

Figure 16. Confusion matrix of ConvNeXt

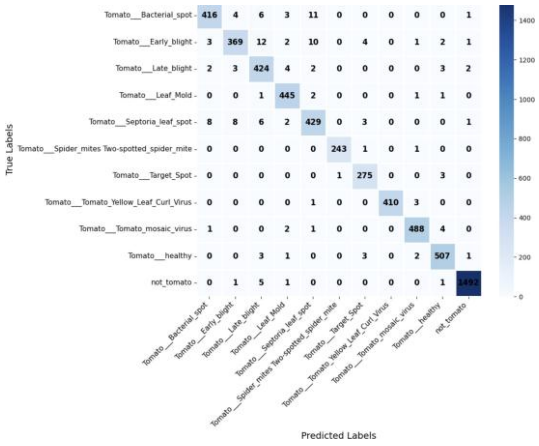

Figure 17. Confusion matrix of ViT

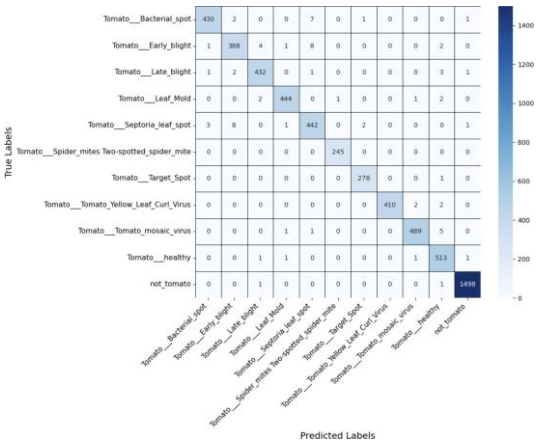

Figure 18:Confusion of Swin

4.6 Quantative Analysis

Table 2. Performance Comparison of models

MODEL	BEST VALIDATION ACCURACY	Precision	RECALL	F1-SCORE	Best Validation Loss	PARAMETERS	APPROX. GFLOPS
ViT	97.43	0.97	0.97	0.97	0.1043	86,397,707	33.73
CNN	87.90	0.89	0.88	0.88	0.3889	25,786,443	1.06
ConvNeXt	97.66	0.97	0.97	0.97	0.1000	87,577,739	30.74
Swin Transformer	98.69	0.99	0.99	0.99	0.0753	27,527,813	4,49

5. Discussion

The present study shows that tomato leaf disease classification should not be evaluated only by overall accuracy. For practical deployment, generalization behavior, discrimination of visually similar diseases, computational efficiency, and robustness to non-target inputs are also important. Among the four evaluated models, Swin Transformer achieved the best overall performance. Swin works by dividing the image into small windows and applying attention inside each window. This helps it focus on local disease details such

as spots, edges, and lesion texture. Then it shifts the windows so information can pass between nearby regions. Because of this, it learns both fine details and wider leaf patterns. This makes Swin better at separating visually similar tomato leaf diseases than the other models. Swin obtained the highest validation accuracy and showed the clearest separation among difficult classes such as Early blight, Septoria leaf spot, Bacterial spot, and Late blight. ViT showed the most stable learning behavior, with closely aligned training and validation curves. ConvNeXt also performed competitively, but required higher computational resources and showed more validation fluctuation. The custom CNN was the fastest model, but its lower accuracy and higher confusion among similar diseases limited its reliability.

However, the results should be interpreted with caution. The models used different input sizes, batch sizes, and fine-tuning settings, so some performance differences may be affected by training configuration. The HSV preprocessing was employed based on prior studies demonstrating its effectiveness in improving leaf segmentation and classification performance. However, no ablation study was conducted in this work to quantify its independent contribution. Similarly, the not_tomato class improves rejection of unrelated inputs, but it is not a complete open-set solution because unknown real-world inputs may still be misclassified. Also, real-field testing was preliminary and conducted on a limited number of government-provided and real tomato leaf images. Therefore, larger field validation is still required. Future work should include field testing, preprocessing ablation, and repeated experiments with statistical validation.

6. Novel Findings and Contributions

This study makes several contributions. First, it provides a comparative evaluation of four distinct deep-learning model families—custom CNN, ConvNeXt, ViT, and Swin Transformer—within a single tomato leaf disease pipeline and on a shared 11-class benchmark. Second, it extends beyond a standard closed-set disease classifier by including a not_tomato rejection class, thereby improving deployment realism. Third, it reduces over-reliance on headline accuracy by jointly examining training and validation curves, validation loss behavior, precision, recall, F1-score, confusion structure and computational cost. Fourth, it supports a more nuanced comparative conclusion than a simple leaderboard: Swin Transformer emerged as the strongest overall model and the best classifier of visually similar diseases, whereas ViT showed the cleanest overfitting and generalization profile. Finally, it identifies exactly where residual difficulty remains, with Early blight, Septoria leaf spot, Bacterial spot, and Late blight forming the hardest lesion-similar cluster across models. At the same time, not_tomato and several visually distinctive classes were comparatively easier.

7. Conclusion

This study compared CNN, ConvNeXt, ViT, and Swin Transformer on an 11-class tomato leaf dataset with 31,042 training and 5,643 validation images. Swin achieved the best overall performance, with 98.69% validation accuracy and the strongest separation of visually similar diseases. ViT showed slightly lower accuracy but the most stable generalization, with closely aligned training and validation curves. ConvNeXt remained competitive but was slower and more oscillatory, while the custom CNN was the fastest but least reliable on difficult classes. Overall, the results show that model performance should be evaluated not only by accuracy, but also by generalization, error patterns, and efficiency. By combining cross-architecture comparison, non-target handling through not_tomato, and detailed performance analysis, this study provides a more practical and realistic evaluation than accuracy-focused approaches.

8. Future Work

Future work should use more standardized training settings across backbones, test the models on larger real-field datasets, perform detailed class-wise evaluation and include an ablation study comparing raw and HSV-preprocessed images.

Acknowledgement

We would like to express our gratitude to everyone who helped us complete this project. We especially extend our heartfelt thanks to Kantipur Engineering College for providing the necessary resources and support throughout this project.

References

[1] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," in ICLR, 2021.

[2] Z. Liu et al, "Swin Transformer: Hierarchical vision transformer using shifted windows," in ICCV, 2021, pp. 9992-10002.

[3] Z. Liu et al., "A ConvNet for the 2020s," in CVPR, 2022, pp. 11966-11976.

[4] N. Srivastava et al., "Dropout: A simple way to prevent neural networks from overfitting," JMLR, vol. 15, pp. 1929-1958, 2014.

[5] J. Chen et al., "Using deep transfer learning for image-based plant disease identification," Comput. Electron. Agric, vol. 173, 2020.

[6] I. Pacal et al., "A systematic review of deep learning techniques for plant diseases," Artif. Intell. Rev, vol. 57, no. 11, 2024.

[7] E. Hamuda et al., "Automatic crop detection under field conditions using the HSV colour space and morphological operations," Comput. Electron. Agric, vol. 133, pp. 97-107, 2017.

[8] G. Waldamichael et al., "Coffee disease detection using a robust HSV color-based segmentation and transfer learning for use on smartphones," Int. J. Intell. Syst., vol. 37, no. 8, pp. 4967-4993, 2022.