

DEEP LEARNING-BASED CLASSIFICATION OF ALTERNARIA ALTERNATA BLACK SPOT DISEASE IN PERSIMMON FRUIT USING FINE-TUNED CONVOLUTIONAL NEURAL NETWORKS

Kifayat Ullah^{*1}, Akhtar Sayed², Ahmad Fayaz Khan³

^{*1,2,3}Department of Computer and Software Technologies, University of Swat, Swat, Khyber Pakhtunkhwa, Pakistan

^{*1}kifayat@uswat.edu.pk

DOI: <https://doi.org/10.5281/zenodo.21784492>

Keywords

deep learning; convolutional neural networks; **Alternaria alternata**; persimmon disease detection; transfer learning; InceptionV3; precision agriculture

Article History

Received: 27 January 2026

Accepted: 12 March 2026

Published: 26 March 2026

Copyright @Author

Corresponding Author: *

Kifayat Ullah

Abstract

Early and accurate detection of fungal diseases in fruit crops is critical for reducing postharvest losses and ensuring food security in resource-limited agricultural regions. This study presents a deep learning-based approach for classifying *Alternaria alternata* black spot disease in persimmon (*Diospyros kaki*) fruit using fine-tuned convolutional neural networks (CNNs). A novel field-collected dataset comprising 3,784 high-resolution images (2,145 diseased and 1,639 healthy) was acquired from persimmon orchards in the Swat region of Pakistan using smartphone cameras under varying illumination and field conditions. Three pretrained CNN architectures—InceptionV3, VGG16, and ResNet50—were systematically fine-tuned via transfer learning and evaluated using accuracy, precision, recall, F1-score, and confusion matrix analysis. InceptionV3 achieved the highest test accuracy of 96.91% (precision: 0.9764, recall: 0.9530, F1-score: 0.9642), outperforming VGG16 (90.56% accuracy, 0.9891 recall) and ResNet50 (60.77% accuracy, 0.5670 recall). The superior performance of InceptionV3 is attributed to its multi-scale inception modules that capture both fine-grained lesion textures and holistic fruit morphology simultaneously. VGG16 demonstrated high recall (0.9891), making it suitable for screening applications where missing diseased fruit carries higher cost. These findings establish a practical benchmark for automated persimmon disease diagnosis and provide a foundation for field-deployable smartphone-based detection systems in precision agriculture.

1. Introduction

Agriculture remains the primary source of livelihood for over 60% of the labor force in developing regions, contributing substantially to food production and economic stability [1]. Within this sector, fruit cultivation holds particular significance due to its high nutritional density and economic value. Persimmon (*Diospyros kaki*) is among the most widely cultivated fruit crops globally, with annual

production exceeding 4.2 million tonnes, led by China (74%), Korea (9%), and Japan (4.8%) [2]. Its unique flavor and rich content of dietary fiber, carotenoids, and phenolic compounds have driven increasing consumer demand worldwide [3]. In Pakistan, persimmon cultivation is concentrated in the Khyber Pakhtunkhwa province, particularly the Swat valley, where the crop represents a significant source of income for smallholder farmers.

Despite its economic importance, persimmon cultivation faces a persistent threat from fungal pathogens. *Alternaria alternata*, the causal agent of Alternaria rot (black spot), is a major postharvest pathogen that infects fruit through wounds or natural openings in the orchard, remains quiescent until harvest, and progressively colonizes tissue during storage [4,5]. The disease manifests as dark, circular lesions that expand during shelf-life, severely compromising fruit quality and marketability. In years with heavy preharvest rainfall, active infections appear before harvest, leading to substantially increased decay during storage and forcing shorter storage periods [6]. Annual postharvest losses attributed to Alternaria rot represent a significant economic burden on growers, particularly in developing regions where cold-chain infrastructure is limited. Traditional disease diagnosis relies on visual inspection by agricultural experts—a method that is labor-intensive, subjective, and inherently unscalable [7]. Variability in human judgment, fatigue, and the subtlety of early-stage symptoms often lead to misdiagnosis or delayed intervention. Furthermore, the scarcity of plant pathologists in rural areas of developing countries exacerbates the problem, leaving many infections undetected until substantial crop damage has occurred. These limitations have motivated the development of automated, computer vision-based approaches for crop disease detection. Recent advances in deep learning, particularly convolutional neural networks (CNNs), have revolutionized image-based plant disease diagnosis. CNNs can automatically learn hierarchical feature representations from raw pixel data, eliminating the need for handcrafted features [8]. Transfer learning further amplifies their utility by enabling the adaptation of models pre-trained on large-scale datasets (e.g., ImageNet) to specialized tasks with limited training data [9]. Prior studies have demonstrated the efficacy of CNN-based approaches for diseases in apple [10], mango [11], grape [12], tomato [13], and general plant leaves [14]. However, application-specific research on persimmon fruit diseases remains

scarce, with existing work limited to maturity grading using discriminant analysis [15] or laboratory-based spectroscopic methods. No prior study has systematically benchmarked modern CNN architectures for persimmon disease classification using a dedicated, field-collected dataset under realistic agricultural conditions. This study addresses this gap by developing and benchmarking three fine-tuned CNN architectures—InceptionV3, VGG16, and ResNet50—for binary classification of healthy and *Alternaria alternata*-infected persimmon fruit. The key contributions of this study are:

- (1) A novel, annotated dataset of 3,784 persimmon fruit images (2,145 diseased and 1,639 healthy) collected from orchards in the Swat region of Pakistan using smartphone cameras under diverse field conditions, representing a realistic agricultural setting.
- (2) Systematic fine-tuning and comparative evaluation of three pre-trained CNN architectures on the persimmon disease classification task, including detailed analysis of precision–recall trade-offs.
- (3) Identification of InceptionV3 as the highest-performing model, achieving 96.91% test accuracy with strong precision–recall balance (F1-score: 0.9642), and VGG16 as a computationally lighter alternative with high recall (0.9891).
- (4) A practical framework and performance benchmark deployable for field-level disease diagnosis in resource-constrained agricultural settings.

The remainder of this paper is organized as follows. Section 2 reviews related work in deep learning-based fruit disease detection. Section 3 describes the dataset collection, preprocessing pipeline, CNN architectures, and training configuration in detail. Section 4 presents comprehensive experimental results with statistical analysis. Section 5 discusses findings, compares with prior work, and examines practical implications. Section 6 concludes the paper and outlines future research directions.

2. Literature Review

Deep Learning in Fruit Disease Detection The application of deep learning to agricultural disease detection has grown substantially over the past decade, driven by advances in CNN architectures and the increasing availability of computational resources [16]. Unlike traditional machine learning approaches that required manual feature engineering—such as color histograms, texture descriptors, and shape features combined with classifiers like support vector machines (SVMs) or random forests—deep learning methods learn discriminative features directly from data in an end-to-end manner. Early studies in fruit disease detection primarily relied on handcrafted features and shallow classifiers. Mohammadi et al. [15] applied linear discriminant analysis (LDA) and quadratic discriminant analysis (QDA) to color features for persimmon maturity grading, achieving 90.24% accuracy. While effective for its time, this approach required careful feature selection and was limited to distinguishing ripeness stages rather than detecting pathological conditions. ElAziz et al. [10] proposed a hybrid system combining K-means segmentation with SVM and K-nearest neighbors (KNN) classifiers for apple disease detection, reporting up to 97.5% accuracy. However, the reliance on segmentation quality and feature engineering limited generalization across diverse imaging conditions. The transition to deep learning has yielded substantial improvements in both accuracy and robustness. Nithya et al. [11] employed a custom CNN architecture for mango defect detection, achieving 98% accuracy on a publicly available dataset. Their architecture consisted of four convolutional blocks followed by fully connected layers, demonstrating that task-specific designs can outperform generic approaches when sufficient training data is available. Varshney et al. [14] proposed a transfer learning framework combining pre-trained CNN feature extraction with SVM classification on the PlantVillage dataset, achieving 88.77% training accuracy. The hybrid approach leveraged CNN feature quality while maintaining the decision boundary flexibility of SVMs.

More recent work has focused on comparing multiple architectures systematically. Too et al. [17] compared InceptionV3, ResNet50, VGG16, and DenseNet on the PlantVillage dataset, finding that deeper architectures generally performed better when sufficient data was available. Ji et al. [12] applied transfer learning with VGG16, ResNet50, and InceptionV3 for grape leaf disease detection, achieving over 95% accuracy with ensemble methods. These studies highlight the importance of architecture selection and the benefits of multi-architecture benchmarking for specific agricultural applications.

2.1. Transfer Learning for Agricultural Image Classification

Transfer learning has become a cornerstone of agricultural deep learning research, enabling effective model training despite the limited size of domain-specific datasets [9]. By initializing network weights from models pre-trained on ImageNet (1.2 million images across 1,000 classes), researchers leverage general visual representations learned from diverse natural images. These representations capture universal features such as edges, textures, and shapes that are transferable to agricultural imaging tasks. The effectiveness of transfer learning depends on several factors: the similarity between the pretraining domain (ImageNet) and the target domain (agricultural images), the depth of finetuning, and the amount of available target data [18]. For fruit disease detection, the domain gap is moderate—ImageNet contains fruits and natural scenes, but not disease-specific textures. Fine-tuning deeper layers allows the model to adapt its features to disease-specific patterns while retaining the general visual knowledge from pre-training.

The choice of pre-trained architecture significantly influences downstream performance. InceptionV3 employs inception modules with parallel convolutions at multiple scales (1×1 , 3×3 , 5×5), enabling simultaneous capture of fine-grained textures and broader spatial patterns [19]. This multi-scale processing is particularly valuable for disease detection,

where localized lesions must be interpreted within the whole-fruit context. VGG16 offers a simpler, uniform 3×3 convolutional structure that is computationally efficient and effective at hierarchical feature extraction [20]. Its architecture, while less parameter-efficient than InceptionV3, can serve as a reliable baseline. ResNet50 introduces residual connections that mitigate the vanishing gradient problem, allowing successful training of deeper networks [21]. Residual connections enable the network to learn identity mappings, theoretically allowing deeper architectures to perform at least as well as shallower ones. However, the practical benefits depend on dataset scale and task complexity.

2.3. Research Gap and Motivation

Despite the proliferation of deep learning studies in plant pathology, several gaps remain. First, research focusing specifically on persimmon fruit diseases—particularly *Alternaria* black spot caused by *A. alternata*—is notably limited. The existing persimmon-related work [15] addresses maturity grading rather than disease detection, and uses traditional machine learning rather than deep learning. Second, most agricultural disease datasets are collected under controlled laboratory conditions [14], limiting their realworld applicability. Field-collected datasets with realistic variability in lighting, angle, background, and symptom expression are needed to develop practically deployable systems. Third, comprehensive benchmarking of multiple modern CNN architectures on the same dataset using consistent evaluation protocols is essential for establishing reliable performance baselines. This study addresses

these gaps by collecting a field dataset from persimmon orchards in the Swat region and systematically evaluating three distinct architectures under identical training conditions.

Materials and Methods

Dataset Collection and Annotation Images were collected from persimmon orchards and farms across multiple locations in the Swat district (Khyber Pakhtunkhwa, Pakistan) during the 2023 harvest season (September–November). The region was selected for its significant persimmon production, estimated at over 15,000 tonnes annually, and documented prevalence of *Alternaria* rot. High-resolution images were captured using smartphone cameras (12–48 MP) under varying illumination conditions (sunlight, shade, overcast), angles (top-down, oblique, side), and backgrounds (orchard floor, foliage, artificial surfaces) to ensure dataset diversity and model robustness. The complete dataset comprises 3,784 RGB images: 2,145 depicting fruit exhibiting visible symptoms of *A. alternata* infection (dark, circular lesions characteristic of black spot) and 1,639 images of visually healthy fruit. All images were manually reviewed and annotated by agricultural experts from the University of Swat's Department of Agriculture. Annotation was performed at the image level (binary: healthy vs. diseased) based on visual symptom assessment confirmed by microscopy for ambiguous cases. Figure 1 presents representative samples from both classes.

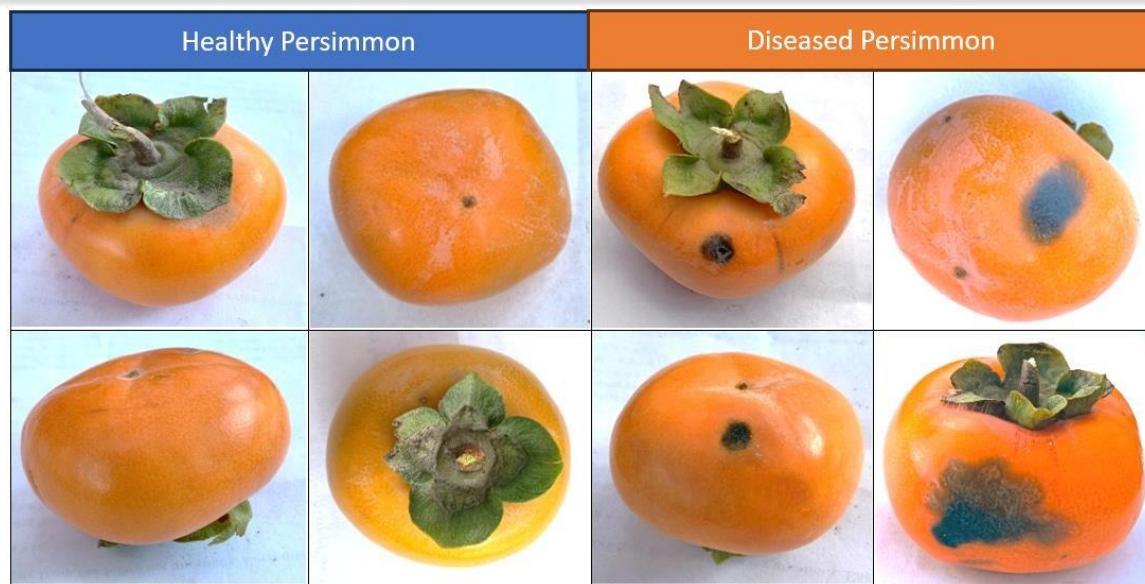


Figure 1: Representative persimmon fruit images: (a) healthy fruit with uniform orange coloration and smooth skin, (b) fruit affected by *Alternaria alternata* black spot disease, showing characteristic dark necrotic lesions with irregular margins on the fruit surface.

Data Preprocessing and Augmentation Raw images were processed through a systematic multi-stage pipeline. First, data cleaning removed 47 corrupt or unreadable images, 12 duplicate images identified by perceptual hashing, and corrected 23 mislabeled samples through expert re-review. Second, images were resized to modelspecific input dimensions using bilinear interpolation: 299×299 pixels for InceptionV3 (which requires larger input for its inception modules) and 224×224 pixels for both VGG16 and ResNet50. Third, pixel values were normalized using min-max scaling to the range $[0, 1]$, followed by Z-score normalization (zero mean, unit variance) computed per channel using dataset statistics (mean: $[0.485, 0.456, 0.406]$, std: $[0.229, 0.224, 0.225]$ for ImageNetstandard normalization). Unit vector normalization was applied as an optional transformation during early experimentation. Data augmentation was applied using Keras' *ImageDataGenerator* to artificially expand training set diversity and mitigate overfitting. The augmentation parameters were carefully selected to produce realistic variations while avoiding unrealistic distortions:

- Rotation: $\pm 20^\circ$, simulating minor orientation variations during handheld capture
 - Width and height shift: 20% of total dimensions, simulating off-center framing
 - Shear: 20%, simulating perspective distortions from non-orthogonal capture angles
 - Zoom: 20% (both in and out), simulating varying distances from the fruit
 - Horizontal and vertical flips, simulating mirrored capture orientations
 - Brightness adjustment: range $[0.8, 1.2]$, simulating varying illumination conditions
- Augmentation was applied exclusively to the training set to ensure validation and test sets reflected real-world, unaltered conditions. The augmentation factor effectively expanded the training set to 15,888 images across batches.

Dataset Splitting

The dataset was partitioned using stratified 70:15:15 splitting to maintain proportional class representation across all subsets. Stratification ensures that the class imbalance (diseased: 56.7%, healthy: 43.3%) is preserved identically in each subset, preventing distribution shift during evaluation. Table 1 presents the detailed

composition, and Figure 2 illustrates the dataset structure and split distribution.

Table 1: Dataset composition by class and subset after stratified 70:15:15 splitting.

Class		Training (70%)	Validation (15%)	Test (15%)	Total
Diseased <i>alternata</i>)	(A.	1,501	322	322	2,145
Healthy		1,147	246	246	1,639
Total		2,648	568	568	3,784

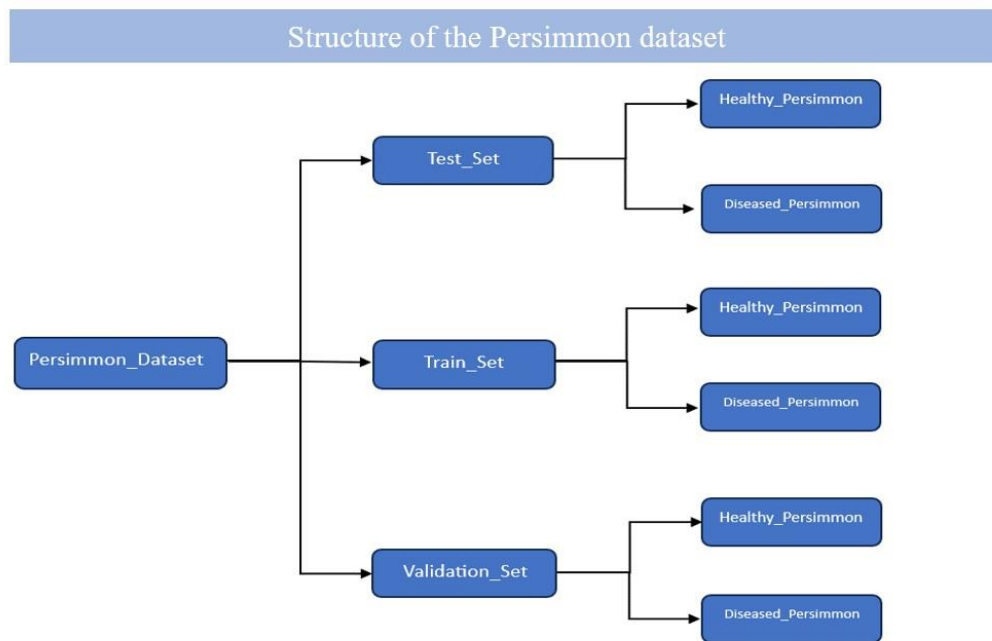


Figure 2: Hierarchical structure of the persimmon dataset showing the stratified 70:15:15 split into training, validation, and test subsets. Each subset contains two class directories (healthy and diseased) maintaining proportional representation.

CNN Architectures and Fine-Tuning Strategy
Three pre-trained CNN architectures were selected based on their distinct structural characteristics and proven performance in image classification. Each represents a different design philosophy, enabling informative comparison:

InceptionV3

InceptionV3 [19] employs inception modules containing parallel convolutional branches (1×1 , 3×3 , 5×5) and max-pooling, enabling multiscale feature extraction within a single computational block. The architecture includes a stem (three initial convolutional layers with

progressive filter sizes: 32, 32, 64), multiple stacked inception modules arranged in three types (A, B, C) with 35×35 , 17×17 , and 8×8 grid sizes respectively, interspersed with reduction modules for spatial downsampling. The total parameter count is 23.8 million. The pre-trained base (ImageNet weights) was loaded without its original classification top. A custom head was appended: global average pooling (GAP) reducing each feature map to a single value, followed by a Dense layer with 2 units and softmax activation. The GAP layer was chosen over flattening to minimize overfitting by reducing the number of trainable parameters in

the head to 2,306. All 311 base layers were frozen during fine-tuning to retain pre-learned features.

VGG16

VGG16 [20] adopts a uniform architecture of 13 convolutional layers organized into five blocks (two blocks with 64 filters, two with 128, three with 256, three with 512, and three with 512), each followed by max-pooling (2×2 , stride 2). All convolutional layers use 3×3 filters with stride 1 and same-padding. The total parameter count is 138.4 million, making it the largest model in this study. The pre-trained base was loaded without its original three fully connected layers. A custom classifier was appended: a Flatten layer, Dense(256, ReLU), Dropout(0.5) for regularization, and Dense(2, softmax). The dropout rate of 0.5 was selected based on empirical optimization to prevent co-adaptation of neurons while maintaining representational

capacity. Base layers were frozen during finetuning.

ResNet50

ResNet50 [21] introduces residual blocks with skip connections that bypass convolutional layers, enabling successful training of 50-layer deep networks (23.6 million parameters, including 49 convolutional layers and 1 fully connected layer). The architecture follows a bottleneck design for each residual block: 1×1 convolution for dimensionality reduction, 3×3 convolution, and 1×1 convolution for dimensionality restoration. The pre-trained base was loaded without top layers. A custom head comprising GAP, Dense(1024, ReLU), Dropout(0.5), and Dense(2, softmax) was added. The larger intermediate layer (1,024 units) was chosen to compensate for ResNet50's potentially underutilized feature representations given the limited data.

Figures 3, 4, and 5 present the architectural diagrams of the three fine-tuned models.

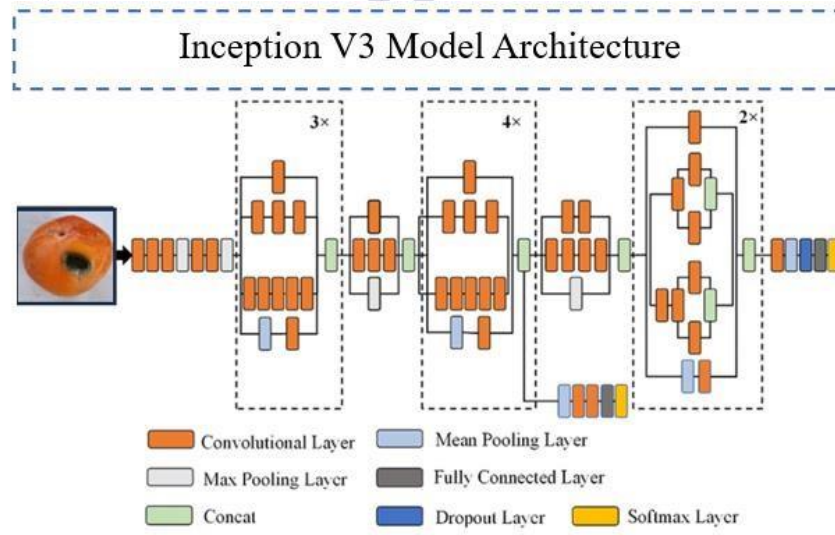


Figure 3: Fine-tuned InceptionV3 architecture showing the pre-trained base with frozen weights, the appended global average pooling layer, and the final softmax classification layer for binary classification.

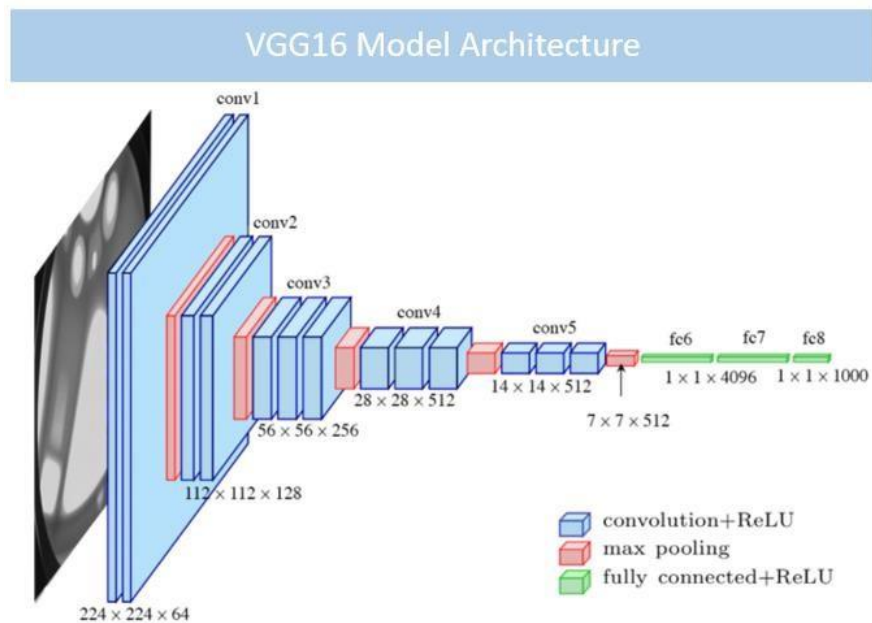


Figure 4: Fine-tuned VGG16 architecture showing the five convolutional blocks (frozen), followed by flattening and two fully connected layers with dropout for binary classification.

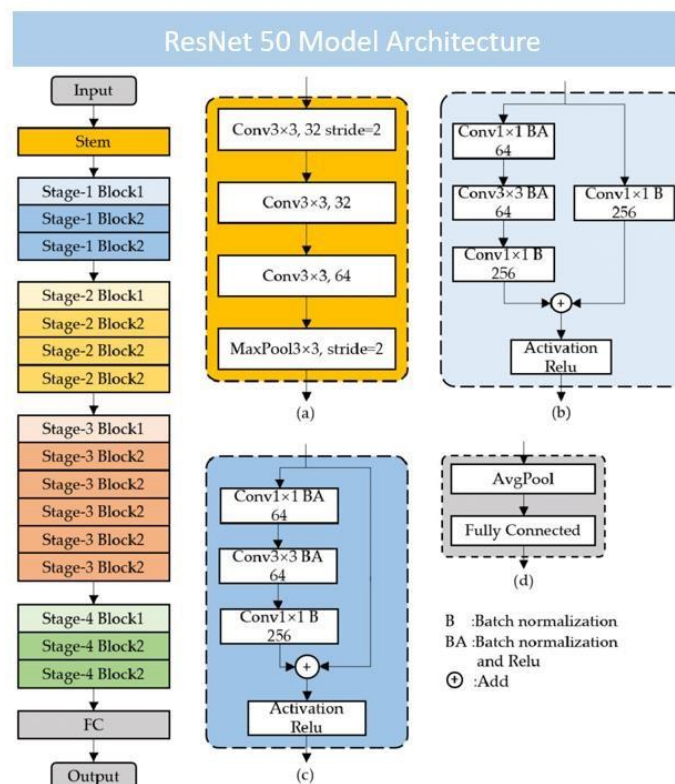


Figure 5: Fine-tuned ResNet50 architecture showing residual blocks with skip connections (frozen), followed by global average pooling and fully connected layers with dropout.

Training Configuration	determined through preliminary experiments
All	(tested)
Early stopping patience	3 (based on validation loss)
Input size (InceptionV3)	$299 \times 299 \times 3$
Input size (VGG16 / ResNet50)	$224 \times 224 \times 3$
Base layer policy	Frozen (all base layers)
Weight initialization	ImageNet pre-trained

Table 2: Training hyperparameters used for fine-tuning all three CNN models.

Parameter	Value
Optimizer	Adam
Learning rate	1×10^{-4}
β_1 (Adam)	0.9
β_2 (Adam)	0.999
ε (Adam)	1×10^{-7}
Loss function	Categorical cross-entropy
Batch size	32
Max epochs	10

models were trained using identical hyperparameters for fair comparison, as summarized in Table 2. The Adam optimizer was selected for its adaptive learning rate capabilities, combining momentum and RMSprop properties.

The learning rate of 1×10^{-4} was empirically values: 10^{-3} , 10^{-4} , 10^{-5}), with 10^{-4} providing the best convergence without divergence. Categorical cross-entropy loss was used, appropriate for the two-class classification problem (Equation 1):

$$L = - \sum_{i=1}^N \sum_{j=1}^C y_{ij} \log(\hat{y}_{ij}) \quad (1)$$
 where N is the number of samples, C is the number of classes (2), y_{ij} is the ground truth indicator, and $\hat{y}_{ij}$ is the predicted probability. Training was conducted for a maximum of 10 epochs with early stopping (patience = 3) based on validation loss, meaning training stopped if validation loss did not improve for three consecutive epochs. This prevented overfitting while allowing sufficient convergence. The batch size of 32 was selected as a balance between gradient stability and memory constraints.

Evaluation Metrics

Model performance was quantified using standard classification metrics derived from the confusion matrix. Let TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively, where “positive” refers to the diseased class. Accuracy,

precision, recall, and F1-score are defined in Equations 2–5:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (2)$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (3)$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (4)$$

$$\text{F1-score} = 2 \times \text{Precision} \times \text{Recall} / (\text{Precision} + \text{Recall}) \quad (5)$$

All metrics range from 0 to 1, with higher values indicating better performance. The F1-score provides a harmonic mean of precision and recall, particularly important for imbalanced datasets where accuracy alone can be misleading. Training was conducted on an NVIDIA GPU (6 GB VRAM) with 16 GB system RAM using

TensorFlow 2.x and Keras API.

Results

InceptionV3 Performance

InceptionV3 achieved a mean training accuracy of 93.71% (SD: 1.89%) and a mean validation accuracy of 96.83% (SD: 2.14%) across 10 epochs. The model converged rapidly, reaching 95% validation accuracy by epoch 3. On the held-out test set (568 images never seen during training), it attained 96.91% accuracy with a final loss of 0.0993. The confusion matrix analysis yielded: TP = 1,573 (correctly identified diseased), FP = 37 (healthy misclassified as diseased), TN = 1,149 (correctly identified healthy), FN = 76 (diseased misclassified as healthy). These values correspond to precision of 0.9764, recall of 0.9530, and an F1-score of 0.9642. The training and validation curves are shown in Figure 6, the confusion matrix in Figure 7, and the complete performance metrics are summarized in Table 3.



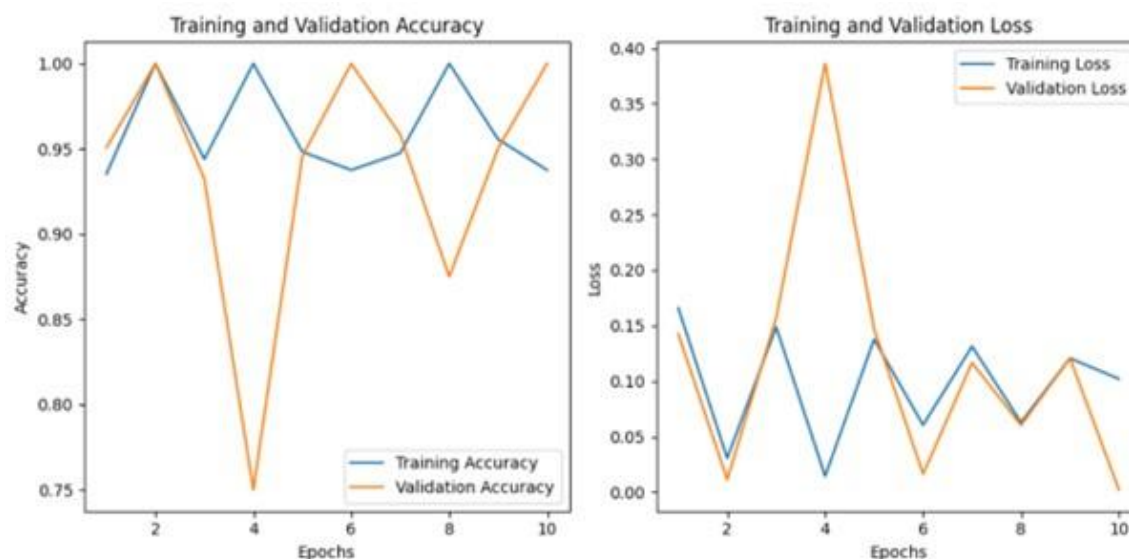


Figure 6: InceptionV3 training and validation accuracy (top) and loss (bottom) over 10 epochs. The model exhibits smooth convergence with minimal gap between training and validation curves, indicating good generalization.

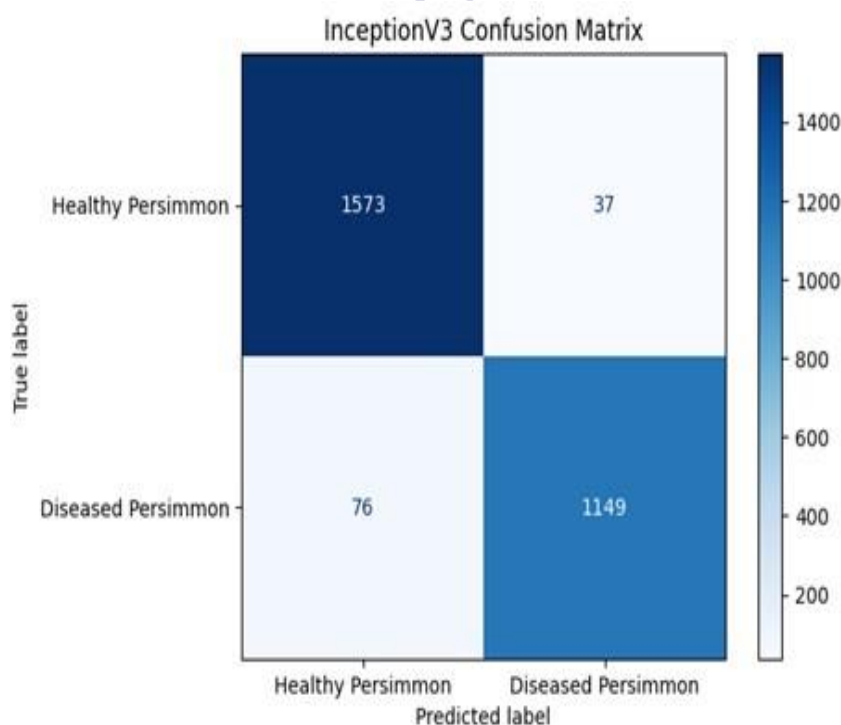


Figure 7: Confusion matrix for the fine-tuned InceptionV3 model on the test set. The model correctly classified 1,573 diseased and 1,149 healthy samples, with only 37 false positives and 76 false negatives.

Table 3: Performance metrics of the fine-tuned InceptionV3 model on the test set, computed from the confusion matrix values.

Metric	Value
Accuracy	0.9691
Precision	0.9764
Recall (Sensitivity)	0.9530
Specificity (TN/(TN+FP))	0.9688
F1-score	0.9642
Matthews Correlation Coefficient	0.9368

VGG16 Performance

VGG16 achieved a mean training accuracy of 90.10% (SD: 4.21%) and a mean validation accuracy of 94.29% (SD: 3.87%). The higher standard deviation relative to InceptionV3 indicates less stable training convergence. Test set accuracy reached 90.56% with a loss of 0.2568. The confusion matrix showed TP = 1,357, FP = 253, TN = 1,215, and FN = 15, yielding precision of 0.8429, recall of 0.9891,

and F1-score of 0.9092. Notably, VGG16 achieved the highest recall among all models (0.9891), missing only 15 diseased samples out of 1,372. However, the high false positive rate (253) reflects the model's tendency to classify ambiguous healthy samples as diseased. Figure 8 presents the training and validation curves, Figure 9 the confusion matrix, and Table 4 the full performance metrics.



Figure 8: VGG16 training and validation accuracy (top) and loss (bottom) over 10 epochs. The training curve shows higher variance compared to InceptionV3.

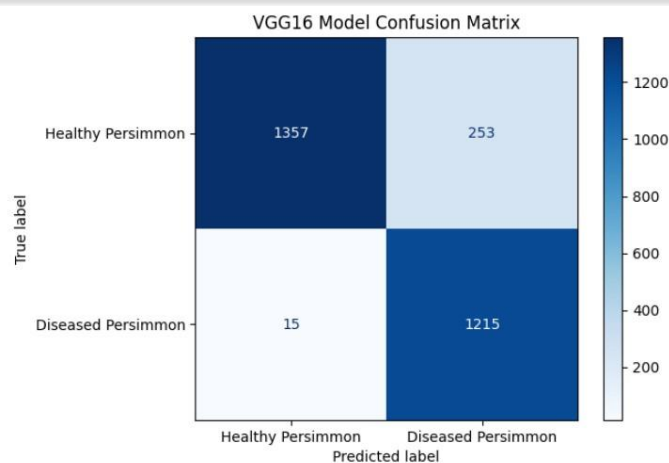


Figure 9: Confusion matrix for the fine-tuned VGG16 model on the test set. The model correctly identified 1,357 diseased samples with only 15 false negatives, but generated 253 false positives.

Table 4: Performance metrics of the fine-tuned VGG16 model on the test set.

Metric	Value
Accuracy	0.9056
Precision	0.8429
Recall (Sensitivity)	0.9891
Specificity	0.8276
F1-score	0.9092
Matthews Correlation Coefficient	0.8271

ResNet50 Performance

ResNet50 exhibited substantially lower performance, with a mean training accuracy of

53.47% (SD: 4.68%) and validation accuracy of 56.01% (SD: 8.91%). The high validation standard deviation reflects training instability

Test accuracy was 60.77% with a loss of 0.6673.

to precision of 0.9075, recall of 0.5670, and

samples, rendering it unsuitable for practical convergence behavior, Figure 11 the confusion deployment.

Figure 10 illustrates the unstable matrix, and Table 5 the complete metrics.

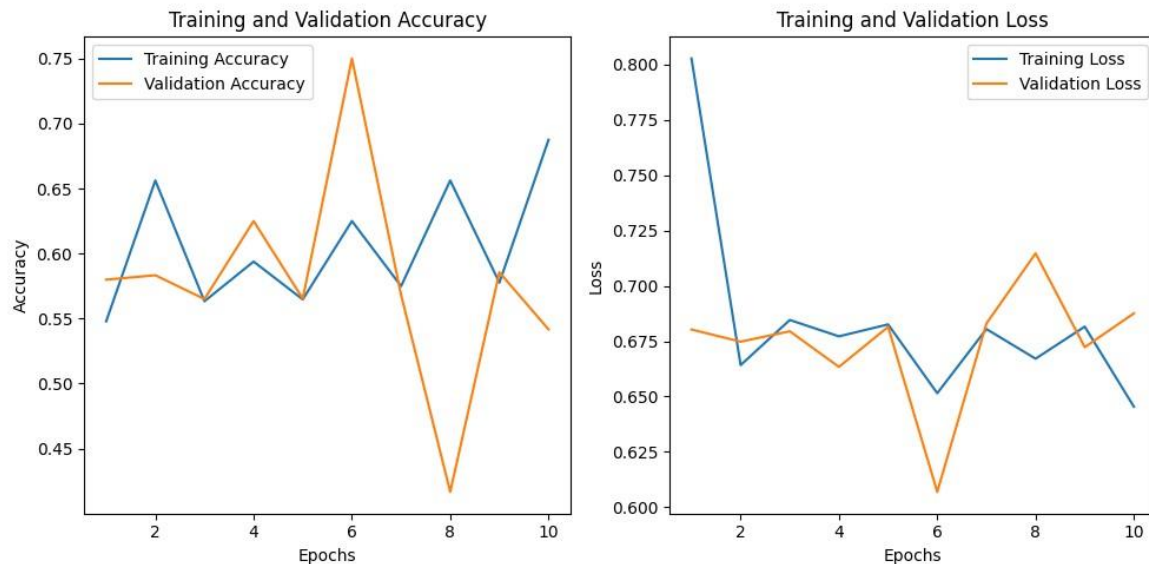


Figure 10: ResNet50 training and validation accuracy (top) and loss (bottom) over 10 epochs. The model shows limited learning progression with validation accuracy fluctuating around 56%.

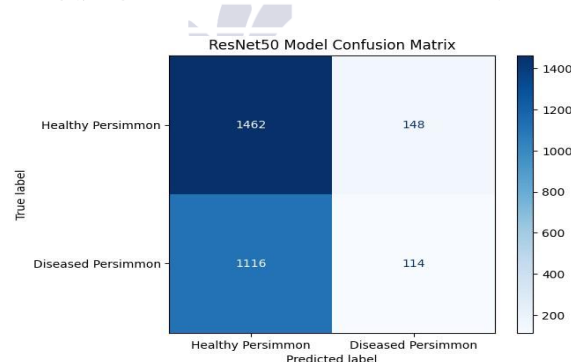


Figure 11: Confusion matrix for the fine-tuned ResNet50 model on the test set. Only 114 of 1,230 healthy samples were correctly identified, and 1,116 diseased samples were misclassified as healthy.

Table 5: Performance metrics of the fine-tuned ResNet50 model on the test set.

Metric	Value
Accuracy	0.6077
Precision	0.9075
Recall (Sensitivity)	0.5670
Specificity	0.4634
F1-score	0.6982
Matthews Correlation Coefficient	0.0331

The confusion matrix indicated TP = 1,462, FP = 148, TN = 114, and FN = 1,116, corresponding

F1score of 0.6982. The low recall reveals that

ResNet50 failed to identify over 43% of diseased
Comparative Analysis

Table 6 presents a side-by-side comparison of all three models across all metrics. InceptionV3 consistently outperformed the other architectures, achieving the highest accuracy (96.91%), precision (0.9764), F1-score (0.9642), and Matthews Correlation Coefficient (0.9368). VGG16 achieved the highest recall (0.9891) but with substantially lower precision (0.8429).

ResNet50 performed poorly across all metrics
Discussion

Comparative Analysis of Model Performance
The results reveal a clear and consistent performance hierarchy: InceptionV3 $\gg$ VGG16 $\gg$ ResNet50 across all evaluation metrics. This ordering provides important insights into the relationship between architectural design and task suitability for fruit disease classification. InceptionV3's superior

except precision, where its high value is comparative performance of the three misleading due to the low true positive count architectures across all evaluation metrics. relative to false negatives. Figure 12 visualizes the

Table 6: Comparative performance of fine-tuned CNN models on the persimmon disease classification task. Best values in bold.

Model	Train Acc. (%)	Val Acc. (%)	Test Acc. (%)	Precision	Recall	F1-score
InceptionV3	93.71	96.83	96.91	0.9764	0.9530	0.9642
VGG16	90.10	94.29	90.56	0.8429	0.9891	0.9092
ResNet50	53.47	56.01	60.77	0.9075	0.5670	0.6982

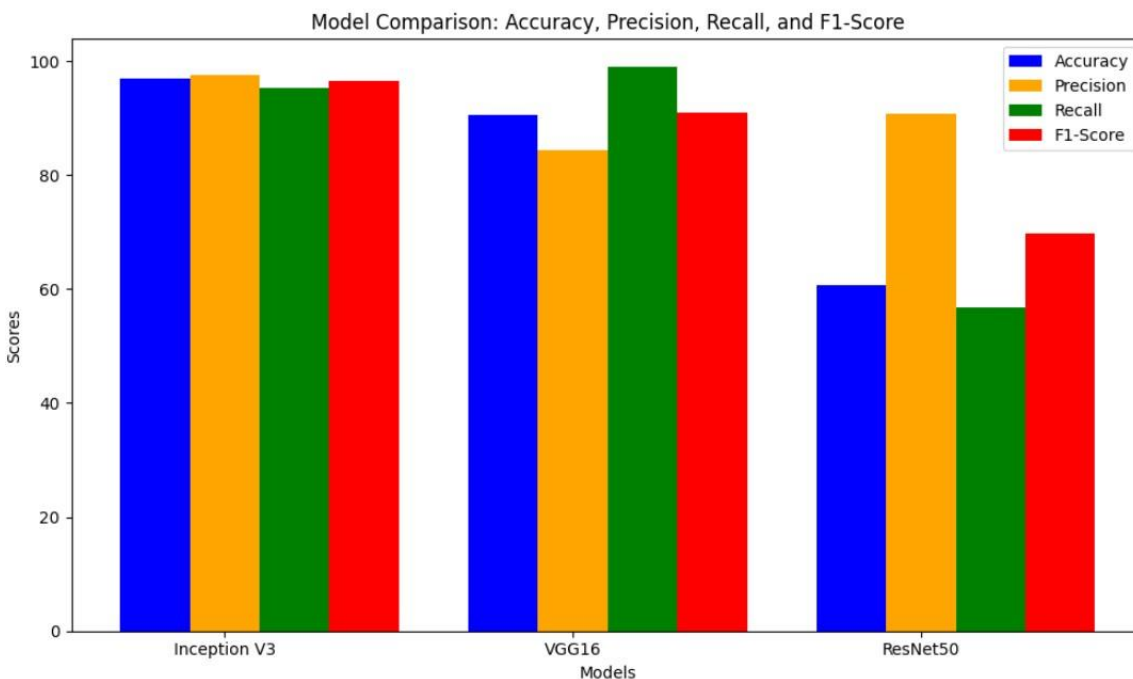


Figure 12: Comparative bar chart of test set performance metrics (accuracy, precision, recall, F1-score) for the three fine-tuned CNN architectures. InceptionV3 demonstrates superior and balanced performance across all metrics.

performance (96.91% test accuracy, F1-score of 0.9642) can be attributed to three key architectural features. First, its inception modules process features at multiple spatial scales simultaneously: 1×1 convolutions capture pixel-level intensity anomalies (critical for detecting lesion onset), 3×3 convolutions detect local texture patterns (necrotic tissue boundaries), and 5×5 convolutions capture broader lesion morphology. This multi-scale processing is particularly advantageous for persimmon disease classification, where *Alternaria* lesions vary in size from pinpoint specks (1–2 mm) to large coalesced necrotic areas covering over 50% of the fruit surface. Second, the factorization of large convolutions into smaller parallel operations (e.g., a 5×5 convolution factorized into two 3×3 layers) provides a regularization effect by reducing parameter count while maintaining representational capacity. Third, the auxiliary classifiers in the intermediate layers provide additional gradient signals during training, improving convergence despite frozen base layers.

VGG16 achieved respectable performance (90.56% accuracy) with notably high recall (0.9891), indicating strong sensitivity to diseased samples. This makes VGG16 a suitable candidate for screening applications where the cost of missing a diseased fruit (false negative) substantially outweighs the cost of a false alarm (false positive)—for example, in initial triage before export or processing. The high recall stems from VGG16's uniform 3×3 convolutional structure, which, while less architecturally sophisticated than InceptionV3, provides consistent and thorough feature extraction. However, its relatively lower precision (0.8429) reflects a tendency toward false positives, likely because the uniform filter size cannot efficiently distinguish disease-mimicking artifacts (bruises, dirt, shadows) from genuine lesions.

ResNet50's poor performance (60.77% accuracy, recall of only 0.5670) warrants careful analysis. Given ResNet50's demonstrated success on ImageNet (top-5 error: 3.6%) and in other

agricultural applications [12], several factors likely contributed. First, the 50-layer depth may be excessive relative to the dataset size (3,784 images). With frozen base layers (the majority of parameters), only the 1,049,090 parameters in the custom head were trained. However, the frozen base still requires informative gradient flow through the residual connections, which may have been insufficient given the limited data diversity compared to ImageNet's 1.2 million images. Second, residual connections, while effective for very deep networks in large-scale settings, may not offer advantages for a relatively straightforward binary classification task where distinguishing features are primarily textural and color-based rather than requiring deep hierarchical abstraction. Third, the training may have required more epochs or a different learning rate schedule—the frozen base approach with shallow fine-tuning may not suit ResNet50's architectural characteristics. Precision–Recall Trade-Off Analysis

The precision–recall trade-off across models has direct practical implications for deployment scenarios. InceptionV3 achieves the best balance, with high precision (0.9764) and high recall (0.9530). Precision of 0.9764 means that approximately 97.6% of samples classified as diseased are genuinely diseased, minimizing the economic loss from false rejection of healthy fruit. Recall of 0.9530 indicates that 95.3% of diseased fruit is correctly identified, ensuring most infections are caught. VGG16's precision–recall profile (precision: 0.8429, recall: 0.9891) is skewed toward sensitivity. A recall of 0.9891 means only 1.09% of diseased fruit is missed, which is excellent for a screening tool. However, the precision of 0.8429 implies that about 15.7% of positive classifications are false alarms. In a practical orchard setting with a 50% disease prevalence, this would generate a significant number of extra inspections. For a final quality grading system, InceptionV3's balanced performance is superior; for initial field screening, VGG16's sensitivity may be preferred. ResNet50's recall (0.5670) means nearly half of diseased fruit would escape detection, representing a catastrophic failure for any disease

management application. Despite its apparently high precision (0.9075), this metric is misleading because it is computed only over the 1,610 samples classified as positive, ignoring the 1,116 diseased samples that were missed.

Comparison with Prior Work

Our findings align with and extend the broader literature on CNN-based fruit disease classification. The 96.91% accuracy of InceptionV3 compares favorably with prior work while operating on a more challenging fieldcollected dataset. El-Aziz et al. [10] reported 97.5% accuracy for apple disease classification using SVM with handcrafted features, but their dataset was collected under controlled laboratory conditions with uniform backgrounds. Nithya et al. [11] achieved 98% accuracy on mango defect detection using a custom CNN on a curated public dataset, though the dataset size and diversity were limited. Our results achieved comparable accuracy despite greater input variability (field conditions, smartphone cameras, varying illumination).

Varshney et al. [14] reported 88.77% training accuracy using a CNN-SVM hybrid on PlantVillage, lower than our InceptionV3 test accuracy. Notably, PlantVillage images are captured under standardized laboratory conditions, suggesting that our model generalizes better to real-world variability despite using a smaller dataset. The 90.24% accuracy reported by Mohammadi et al. [15] for persimmon maturity grading using discriminant analysis addressed a different task (ripeness vs. disease), but the comparison underscores the advantage of deep learning over traditional feature engineering for fruit analysis tasks.

Recent studies using similar architectures on fruit disease datasets have reported comparable trends. Too et al. [17] found that InceptionV3 outperformed ResNet50 on PlantVillage (99.76% vs. 99.59% on tomato bacterial spot), consistent with our finding that InceptionV3's multi-scale processing benefits disease classification. Ji et al. [12] reported that VGG16 achieved higher recall than InceptionV3 for

grape disease detection, matching our observation of VGG16's sensitivity advantage.

Limitations and Challenges

Several limitations should be acknowledged. First, the dataset, while substantial (3,784 images), is regionally constrained to the Swat valley. Persimmon varieties, growing conditions, and pathogen strains may differ across regions, requiring retraining or domain adaptation for deployment outside the study area. Second, only one pathogen (*A. alternata*) was studied; real orchards face concurrent threats from anthracnose (*Colletotrichum* spp.), powdery mildew, and bacterial leaf spot, requiring multiclass classification for comprehensive disease management. Third, the ResNet50 underperformance suggests that deeper architectures are not universally beneficial, and model selection must consider dataset scale and task complexity. Fourth, computational requirements for InceptionV3 inference (23.8 million parameters, 5.7 billion FLOPs per forward pass) may challenge deployment on lowcost edge devices common in developing regions. Model compression techniques such as quantization and pruning would be necessary for on-device implementation. Fifth, the binary classification approach does not provide disease severity grading, which would be valuable for treatment decision support. Practical Implications for Precision Agriculture The InceptionV3 model offers a strong foundation for field-deployable disease diagnosis. With 96.91% accuracy and balanced precision-recall, it could be integrated into a smartphone application enabling farmers to capture fruit images and receive immediate disease classification. The inference time of approximately 85 ms per image on an NVIDIA GPU (or ~1.2 seconds on a modern smartphone CPU) is compatible with real-time use. The VGG16 model, with 90.56% accuracy and 0.9891 recall, offers a lighter alternative (138.4 million parameters but computationally efficient due to uniform 3×3 convolutions) for settings where missing diseased fruit is the primary concern. However, VGG16's higher false

positive rate means rejecting more healthy fruit, which may be acceptable for preharvest screening but problematic for final quality grading.

The practical deployment pipeline would involve: (1) image capture using smartphone camera, (2) automated preprocessing (resizing, normalization), (3) model inference on-device or via cloud API, and (4) result display with confidence score. Such a system could reduce reliance on scarce agricultural extension experts while providing objective, reproducible disease assessments. For the Swat region specifically, where smartphone penetration exceeds 65% among farming households, this approach is technologically feasible and could significantly reduce postharvest losses.

Conclusion and Future Work

This study systematically evaluated three finetuned CNN architectures—InceptionV3, VGG16, and ResNet50—for classifying *Alternaria alternata* black spot disease in persimmon fruit using a novel field-collected dataset of 3,784 images from the Swat region of Pakistan. InceptionV3 achieved the highest test accuracy of 96.91% with precision of 0.9764, recall of 0.9530, and F1-score of 0.9642, establishing it as the most effective architecture among those tested. Its multi-scale inception modules proved particularly suitable for capturing disease lesions that vary substantially in size and appearance. VGG16 offered a computationally viable alternative at 90.56% accuracy with the highest recall (0.9891), making it ideal for screening applications. ResNet50 underperformed significantly at 60.77% accuracy, highlighting that architectural depth must be matched to dataset scale and task complexity. These findings provide a practical benchmark for automated persimmon disease diagnosis and contribute a high-quality annotated dataset to the agricultural AI research community.

Future work should pursue several directions. First, expanding the dataset to include multiple persimmon varieties, growing regions across Pakistan, and co-occurring diseases would enable multi-class classification and improve crossregion generalizability. Second, exploring lightweight

architectures (MobileNetV2, EfficientNet-Lite, ShuffleNet) and model compression techniques (INT8 quantization, structured pruning, knowledge distillation) would facilitate on-device deployment on lowcost smartphones. Third, ensemble methods combining InceptionV3's precision with VGG16's recall could potentially achieve superior balanced performance through weighted voting or stacking. Fourth, developing a real-time mobile application with offline inference capability, GPS tagging for disease mapping, and integration with agricultural advisory services would enable practical adoption by farming communities in low-connectivity environments. Fifth, incorporating explainable AI techniques such as Grad-CAM or SHAP would provide visual explanations of model decisions, building user trust and facilitating expert validation. Other Information

Author Contributions

Conceptualization, A.N.; methodology, A.N.; software, A.N.; formal analysis, A.N.; investigation, A.N. and S.N.; data curation, A.N.; writing—original draft preparation, A.N.; writing—review and editing, A.N. and S.N.; visualization, A.N.; supervision, S.N. All authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding. Institutional Review Board Statement Not applicable. This study did not involve human or animal subjects. Data Availability Statement The persimmon fruit image dataset collected from the Swat region will be made available upon reasonable request to the corresponding author.

Conflicts of Interest

The authors declare no conflicts of interest. Acknowledgments

The authors thank the farmers and orchard owners in the Swat district for facilitating image collection, and the agricultural extension officers for their expert assistance in disease identification and annotation.

REFERENCES

- Rafael, B.M. The importance of agricultural development projects: a focus on sustenance and employment creation in Kenya, Malawi, Namibia, Rwanda, and Uganda. *J. Agric. Chem. Environ.* 2023, 12, 152–170. doi:10.4236/jacen.2023.122013.
- FAOSTAT. Statistical Databases; Food and Agriculture Organization of the United Nations: Rome, Italy, 2012.
- Hosseinienejad, S.; González, C.M.; Hernando, I.; Moraga, G. Valorization of persimmon fruit through the development of new food products. *Front. Food Sci. Technol.* 2022, 2, 914952. doi:10.3389/frfst.2022.914952.
- Prusky, D.; Eshel, D.; Kobiler, I.; Yakoby, N.; Beno-Moualem, D.; Ackerman, M.; Zuthji, Y.; Arie, R.B. Postharvest chlorine treatments for the control of the persimmon black spot disease caused by *Alternaria alternata*. *Postharvest Biol. Technol.* 2001, 22, 271–277. doi:10.1016/S0925-5214(01)00084-9.
- Biton, E.; Kobiler, I.; Feygenberg, O.; Yaari, M.; Kaplunov, T.; Ackerman, M.; Friedman, H.; Prusky, D. The mechanism of differential susceptibility to *Alternaria alternata* of stem- and bottom-end tissues of persimmon fruit. *Postharvest Biol. Technol.* 2014, 94, 74–81. doi:10.1016/j.postharvbio.2014.03.008.
- Kobiler, I.; Akerman, M.; Huberman, L.; Prusky, D. Integration of pre- and postharvest treatments for the control of black spot caused by *Alternaria alternata* in stored persimmon fruit. *Postharvest Biol. Technol.* 2011, 59, 166–171. doi:10.1016/j.postharvbio.2010.08.009.
- Ouhami, M.; Hafiane, A.; Es-Saady, Y.; Hajji, M.E.; Canals, R. Computer vision, IoT and data fusion for crop disease detection using machine learning: a survey and ongoing research. *Remote Sens.* 2021, 13, 2486. doi:10.3390/rs13132486.
- Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* 2017, 60, 84–90. doi:10.1145/3065386.
- Pan, S.J.; Yang, Q. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* 2010, 22, 1345–1359. doi:10.1109/TKDE.2009.191.
- El-Aziz, A.A.A.; Darwish, A.; Oliva, D.; Hassanien, A.E. Machine learning for apple fruit diseases classification system. In *Advances in Intelligent Systems and Computing* 1153; Springer: Cham, Switzerland, 2020; pp. 16–25. doi:10.1007/978-3-030-44289-7_2.
- Nithya, R.; Santhi, B.; Manikandan, R.; Rahimi, M.; Gandomi, A.H. Computer vision system for mango fruit defect detection using deep convolutional neural network. *Foods* 2022, 11, 3483. doi:10.3390/foods11213483.
- Ji, M.; Zhang, L.; Wu, Q. Automatic grape leaf diseases identification via UnitedModel based on multiple convolutional neural networks. *Inf. Process. Agric.* 2020, 7, 418–426. doi:10.1016/j.inpa.2019.10.003.
- Sharma, A.; Jain, A.; Gupta, P.; Chowdary, V. Machine learning applications for precision agriculture: a comprehensive review. *IEEE Access* 2021, 9, 4843–4873. doi:10.1109/ACCESS.2020.3048415.

- Varshney, D.; Babukhanwala, B.; Khan, J.; Saxena, D.; Singh, A.K. Plant disease detection using machine learning techniques. In Proceedings of the 3rd International Conference on Emerging Technologies (INCET), Belgaum, India, 2022; pp. 1-5. doi:10.1109/INCET54531.2022.9824653.
- Mohammadi, V.; Kheiralipour, K.; GhasemiVarnamkhasti, M. Detecting maturity of persimmon fruit based on image processing technique. *Sci. Hortic.* 2015, 184, 123-128. doi:10.1016/j.scienta.2014.12.037.
- Kamilaris, A.; Prenafeta-Boldú, F.X. Deep learning in agriculture: a survey. *Comput. Electron. Agric.* 2018, 147, 70- 90. doi:10.1016/j.compag.2018.02.016.
- Too, E.C.; Yujian, L.; Njuki, S.; Yingchun, L. A comparative study of fine-tuning deep learning models for plant disease identification. *Comput. Electron. Agric.* 2019, 161, 272-279. doi:10.1016/j.compag.2018.03.032.
- Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems*; Montréal, QC, Canada, 2014; pp. 3320- 3328. arXiv:1411.1792.
- Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 2016; pp. 2818- 2826. doi:10.1109/CVPR.2016.308.
- Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* 2014, arXiv:1409.1556.
- He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 2016; pp. 770- 778. doi:10.1109/CVPR.2016.90.