Towards Intelligent Disease Phenotyping in Peach: A Deep Feature Extraction Framework for Leaf Disease Detection Under Real Field Conditions

Aditi Mankotia Mankotia
Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir
Showkat Rasool
Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir
Mushtaq Ahmad Bhat
drmushtaqabr@gmail.com
Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir
Haris Nabi Shah
Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir
Mohammad Saqib Dar
Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir
Zahoor Ahmad Bhat
Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir
Ashiq Hussain Pandit
Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir
Mudasir Manzoor Kirmani
Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir
Suhail Quyoom Wani
Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir
Syed Bisma Nisar
Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir
Zuhaib Farooq
Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir
Khalid Zaffar Masoodi
Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir
Nahida Tun Nisa
University of Kashmir
Mujataba Abdullah Kumar
Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir

Article

Keywords: Deep learning, Transfer learning, CNN, Plant Disease Detection, Precision Agriculture

Posted Date: August 24th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10396262/v1

Additional Declarations: No competing interests reported.

Towards Intelligent Disease Phenotyping in Peach: A Deep Feature Extraction Framework for Leaf Disease Detection Under Real Field Conditions

Aditi Mankotia[a], Showkat Rasool[b], Mushtaq Ahmad Bhat*[a], Haris Nabi shah[b], Mohammad Saqib Dar[b], Zahoor Ahmad Bhat[a], Ashiq Hussain Pandit[c], Mudasir Manzoor Kirmani[d], Suhail Quyoom Wani*[a], Syed Bisma Nisar[a], Zuhaib Farooq[e] , Khalid Zaffar Masoodi[f], Nahida Tun Nisa[g], Mujtaba Abdullah Kumar[a]

[a]Division of Plant Pathology, Sher-e Kashmir University of Agricultural Sciences and Technology, Shalimar, 190025, J&K, India
[b]Center for Artificial Intelligence and Machine Learning, CoAE&T, Sher-e Kashmir University of Agricultural Sciences and Technology, Shalimar, 190025, J&K, India
[c]Division of Fruit Science, Sher-e Kashmir University of Agricultural Sciences and Technology, Shalimar, 190025, J&K, India
[d]Division of Social Science, Sher-e-Kashmir University of Agricultural Sciences and Technology, FoF, Rangil 190025, J&K, India
[e]Division of Entomology, Sher-e Kashmir University of Agricultural Sciences and Technology, Shalimar, 190025, J&K, India
[f]Division of Plant biotechnology, Sher-e Kashmir University of Agricultural Sciences and Technology, Shalimar, 190025, J&K, India
[g]Departement of Botany, Beerwah, University of Kashmir, Shalimar, 193411, J&K, India

*Corresponding author email: drmushtaqabr@gmail.com sahilquyoom103@gmail.com

Abstract

In modern agriculture, it is essential to identify the early symptoms of plant diseases and to accurately maintain the productivity of the crop and reduce economic losses. Foliar diseases are a special concern in peach because they can cause yield as well as quality if not timely detected. Artificial intelligence, machine learning and deep learning are some of the advanced technologies that are gaining great importance in today's agriculture, especially with the image analysis applications. In this study, a deep learning method for automated detection and classification of three peach leaf diseases and healthy class was presented based on image data. The dataset were taken at different phenological and disease stages under temperate conditions in Kashmir with four classes Healthy, Leaf Curl, Shot hole and Rust. Three convolutional neural networks (CNNs) architectures were applied, VGG-16, ResNet 50 and Xception were trained using transfer learning and Inception-V4 was trained from scratch for a comparative study of the learning strategies. Data augmentation techniques were applied to improve generalization. Results show that all models were able to learn disease specific features well. The result of Inception-V4 was found to be highest with 97.50%, followed by ResNet-50 with 94.49%, VGG-16 with 92.04% and Xception with 75.95%. The results of transfer learning-based architectures were also good and competitive but the best results obtained from the Inception-V4 architecture reveal its capability in modelling complex visual patterns. The results highlight the potential of deep learning techniques for early detection of diseases in peach, supporting precision agriculture and better disease management.

Keywords: Deep learning, Transfer learning, CNN, Plant Disease Detection, Precision Agriculture

Introduction

Peach (Prunus persica L. Batsch) is one of the most economically important stone fruit species in terms of nutritional value and commercial demand. Peach production is constantly endangered to major foliar diseases like leaf curl, shot hole, rust etc that affect photosynthesis, fruit development and orchard productivity with extra management costs. Early and accurate diagnosis of such diseases is needed for applying timely control measures and keeping peach trees healthy[1].

The most common method for diagnosing diseases in peaches is mainly based on visual observation by the trained grower or agricultural expert. This is useful for small scale but not helpful for large plantations because is time-consuming, labour-intensive and subjective. Furthermore, visual signs of the same disease are common to a wide variety of diseases with different severity level which can lead to misdiagnosis and mismanagement of the disease. The drawback focuses on the importance of developing automated, accurate and scalable disease detection tools that will help growers in real time decision making.

Artificial intelligence (AI) has made significant progress in recent years and is now contributing to the formation of automated diagnostic systems with the aim of rapid and precise disease detection. AI tools are being applied in many sectors such as agriculture, to identify and classify a wide range of diseases[2]. In the field of plant pathology, Convolutional Neural Networks (CNNs) have proven to be an effective tool for image-based pattern recognition and disease classification[3]. These models are user-friendly applications in mobile apps that mark an important advancement, providing smallholder farmers and other non-expert users with reliable diagnostic assistance without requiring expertise in agronomy or phytopathology[4]. The wide availability of AI can have a significant impact on early detection of disease, timely response to issues and more sustainable agricultural practices.

Initially, machine learning methods and manually extracted image features were used for automatic plant disease diagnosis. In general, these methods consisted of two separate stages: feature extraction and classification. Under controlled imaging conditions, Pydipati *et al.*[5] used color texture features and trained an ANN classifier to diagnose citrus disease from leaf images, showing that automated systems can compete with human experts. Rothe and Kshirsagar (2015)[6] used SVM on mango leaf disease classification using morphological and colour segmentation features which resulted in reasonable accuracy but the noted sensitivity of the pipeline to lighting variation and background clutter. With the advent of deep learning, the breakthrough of Krizhevsky *et al.*[7] with the introduction of AlexNet, image recognition tasks shifted to a whole new paradigm. The Convolutional Neural Networks (CNNs) were able to learn hierarchical representations of features, which were task specific, directly from the raw pixel data without the need for any manual feature engineering. One of the most influential early papers that leveraged deep learning for plant disease detection was by Mohanty *et al.*[8]. They fine-tuned AlexNet and GoogLeNet and reached a classification accuracy of over 99% under controlled conditions on the PlantVillage dataset consisting of more than 54,000 images of 26 diseases across 14 crop species. High performance was promising, but later investigations revealed that the models struggle to achieve high accuracy when tested with images taken in the field, highlighting the gap between lab and field. Ferentinos (2018) continued on this research and tested various CNN architectures such as AlexNet, AlexNetOWTBn, GoogLeNet, Overfeat and VGG with the PlantVillage dataset and achieved the highest accuracy results of more than 99.5%, which reconfirmed the ability of deep learning over traditional approaches to classify disease on plants[9]. These results set the baseline for the use of CNNs in future research on agricultural image analysis. To date, the development of plant disease detection based on deep learning has moved from the basic convolutional neural networks (CNNs) to the highly optimised multi-architecture ensembles (MANs). Initial studies demonstrated feasibility of this technology, such as the work by Kawasaki *et al.*[10] who used a CaffeNet architecture to obtain a 94.9% accuracy in the classification of cucumber leaf images and by Dos *et al.*[11] who used a similar model to the classification of images from soybean fields with 99.0% accuracy. As the field grew up, people started getting bigger and bigger data sets, such as Plant Village, to compare performance between different species. Wang *et al.*[12] showed that the VGG and ResNet architectures can attain 90.4% accuracy for 50,000 images, whereas9 extended the study to include 58 plant classes and obtained 99.53% accuracy with architectures such as GoogLeNet. This quest for more accuracy was advanced by Too *et al.* [13], who achieved a high of 99.75% in a dense set of networks, such as DenseNet-121 and Inception-V4.

In the last few years, the scope has become broader with a focus on crop-specific applications and computational efficiency for practical implementation. Rozlan and Hanafi[14] used the images of chili plants and achieved an accuracy of 98.83% using EfficientNetB0, whereas Kumar and Kumar [15] and Howlader *et al.* [16] demonstrated the effectiveness of VGG and deep CNNs for tomato, potato and guava leaves, respectively. Sutaji and Yildiz[17] were able to achieve high accuracy of 99.52% using lightweight networks such as MobileNetV2 and Xception that had not the deep networks exhibited high computations. At the same time, the innovation of the architectural design has resulted in the development of custom-made architectures like the 7-layer CNN developed for seedling classification[18] and the hybrid DCNN-SVM approach that achieved 97.5% accuracy in rice disease detection[19]. These studies represent a path towards more generalized image recognition to more specialized, highly precise tools that can be used to support global agricultural productivity.

While significant advances have been made in the classification of plant disease with deep learning, there are still several key challenges that have yet to be addressed in the diagnosis of peach diseases. Most of the previous works are based on controlled laboratory data or public image repositories like PlantVillage in which images are captured with consistent lighting, background and low environmental noise. These datasets do not necessarily capture the complexity of real orchard environments, as differences in illumination, leaf angle, occlusion, symptom severity and other factors such as background noise can impact model accuracy. In addition, few studies have been conducted on peach leaf disease classification under temperate agro-climatic conditions especially in Kashmir valley where disease epidemiology and symptom expression is significantly different at different phenological stages. Moreover, the evaluation of various deep convolutional neural network architectures in the real field is still not well studied. Hence, the aim of the current study is to fill this gap by designing and testing a deep learning-based approach for peach leaf disease classification using a large number of field-acquired images under natural orchard conditions. The study also quantifies the ability of transfer learning-based architectures compared to the model trained from scratch to learn the disease-specific visual representations by comparing their performances on the test set.

Implementation of the models

The methodology has been designed on the basis of peach leaf disease classification, as illustrated in Fig. 1. Images of the diseased leaves were captured and pre-processed by resizing and normalizing them into a uniform size. Data augmentation was applied to increase diversity and reduce overfitting. Four deep learning architectures, VGG-16, ResNet-50, Xception and Inception-V4 were used for feature extraction and classification. Model performance was evaluated using accuracy, precision, recall and F1-score.

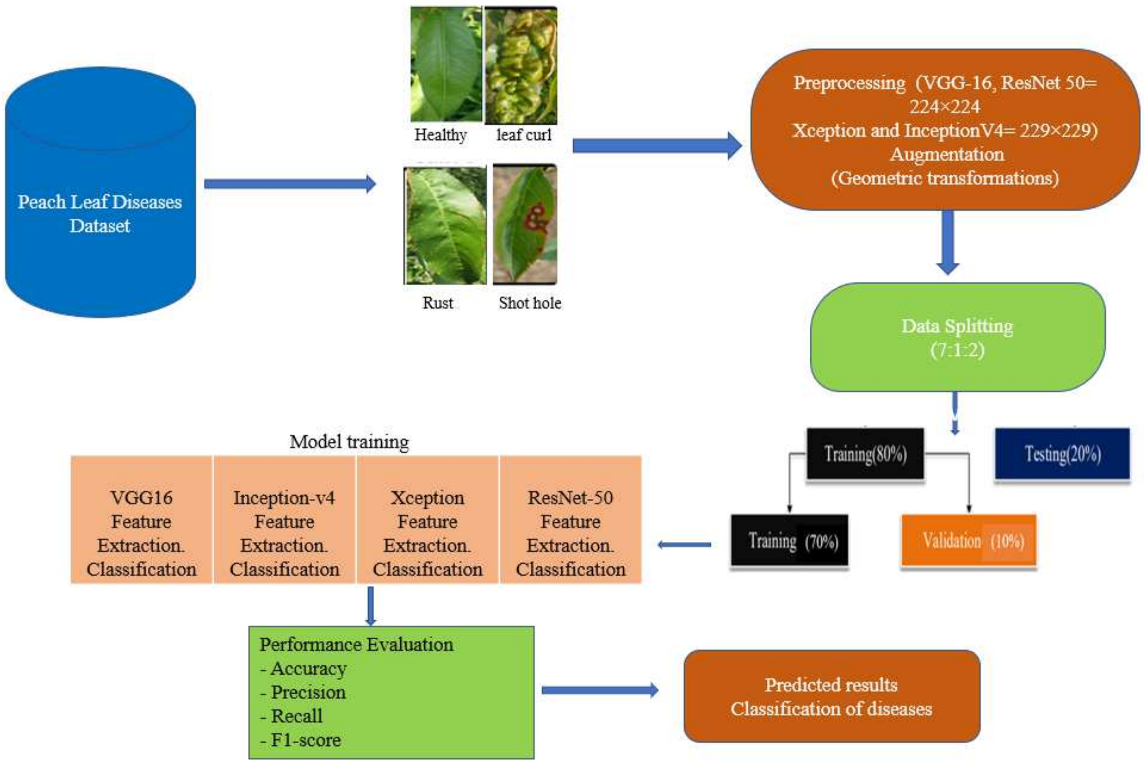

Fig. 1. Overview of the Methodological Workflow

Dataset

The dataset used in this study comprises high-resolution images of peach leaves exhibiting both healthy and diseased conditions, collected under the temperate agro-climatic conditions of Kashmir. The images were categorized into distinct classes based on visible symptomatology corresponding to specific diseases commonly affecting peach plants in the region. Three major diseases of peach included in this study are leaf curl, shot hole and rust, each caused by distinct fungal pathogens and exhibiting unique visual symptoms. Leaf curl, caused by Taphrina deformans, is characterized by thickening and puckering of leaf blades along the midrib, resulting in significant distortion of leaf structure[20]. Shot hole disease caused by Wilsonomyces carpophilus, presents as small purplish-black lesions that enlarge, become necrotic and eventually fall out, producing a shot-hole appearance [21]. Rust, caused by Tranzschelia discolor, is identified by orange to rusty-brown pustules predominantly on the undersides of leaves[22]. Data were collected at distinct phenological stages, corresponding temporally with disease progression stages, to evaluate how plant development influences pathogen progression and how disease severity varies across physiological states. The dataset consists of 10,000 images of Prunus persica L. plants and were collected under an uncontrolled environment with various illuminations, views, distances and from different field environments of Sher-e-Kashmir University of Agriculture Sciences and Technology, Kashmir (SKUAST-K) where they grow different varieties of fruits for educational and research purpose. Other than SKUAST-K, data collection was also carried out in various commercial peach growing areas of the valley like Rangreth. The images were captured by a Sony DSC T-2 digital camera with the dimension resolution of 1080×1920.

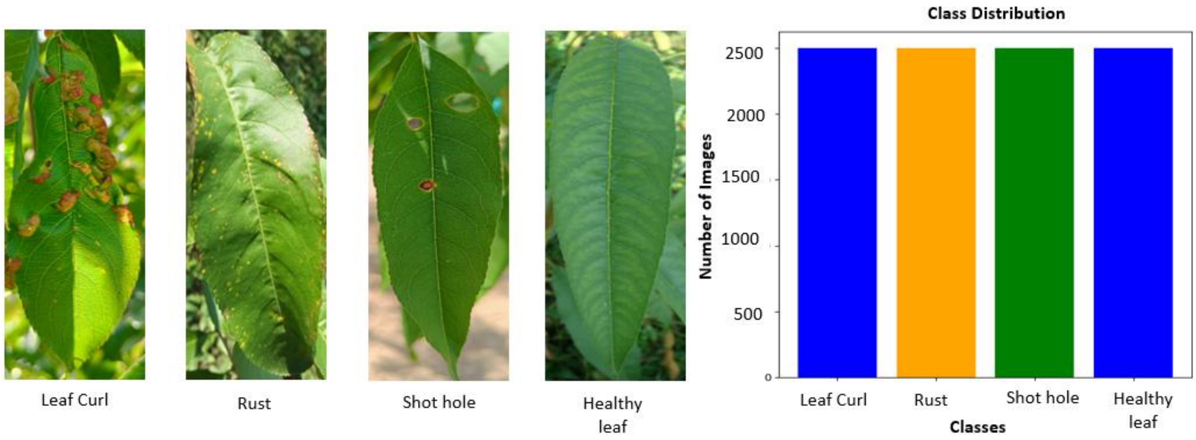

Fig. 2 Sample images and class-wise distribution of peach leaf diseases

Experimental Models

VGG-16

The VGG-16 architecture[23] is a deep convolutional neural network designed for image recognition and consists of 16 layers, including 13 convolutional layers and 3 fully connected layers (Fig. 3). The network takes a 224×224×3 RGB image as input and employs 3×3 convolutional filters with a stride of 1 and 1-pixel padding to preserve spatial resolution. The architecture is organized into five convolutional blocks. The first two blocks contain 64 and 128 filters, respectively, each followed by a 2×2 max-pooling layer with stride 2. The third block includes three convolutional layers with 256 filters, while the final two blocks consist of three convolutional layers each with 512 filters, followed by max pooling. After feature extraction, the network uses two fully connected layers with 4096 neurons each, followed by a final fully connected layer with 10 neurons for classification. A SoftMax activation function produces the output probabilities, while a ReLU activation is applied to all hidden layers.

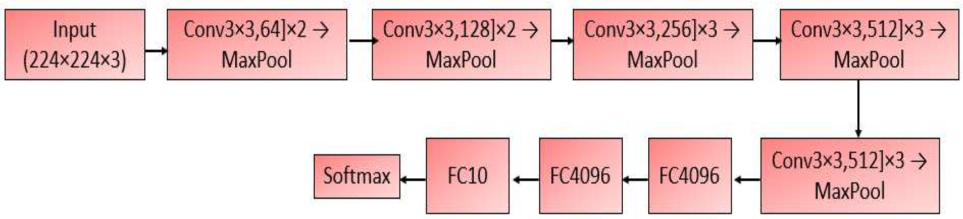

Fig. 3 VGG-16 Model

ResNet-50

The ResNet-50 architecture (Fig. 4), introduced by He *et al.*[24] is a 50-layer deep convolutional network designed for image classification. The network accepts an RGB image of size 224×224×3 as input. Initially, zero padding is applied, followed by a 7×7 convolution with 64 filters and a stride of 2 and a 3×3 max-pooling layer with a stride of 2. The core of the architecture consists of four residual stages. The first stage includes 1×1, 3×3 and 1×1 convolutions with 64, 64 and 128 filters, repeated three times. The second stage uses 128, 128 and 512 filters, repeated four times, while the third stage applies 256, 256 and 1024 filters, repeated six times. The final layer consists of 512, 512 and 2048 filters, three times. These residual blocks are followed by a global average pooling layer that further downsamples in space. Finally, a fully connected layer with 10 neurons and a SoftMax activation function makes the final classification.

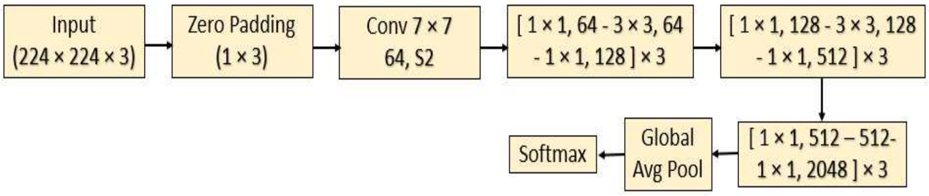

Fig. 4 ResNet-50 Model

Xception

The Xception model [25], also known as the Extreme Inception architecture (Fig. 5), is a deep convolutional network comprising 71 layers, including convolutional, depthwise separable convolutional, pooling and activation layers. The network takes an RGB image of size 299×299×3 as input. The architecture is divided into three main stages: entry flow, middle flow and exit flow.

The entry flow begins with standard convolutional layers using 3×3 kernels with 32 filters, followed by ReLU activation and 2×2 max pooling with stride 2 for spatial downsampling. The defining feature of Xception is its use of depthwise separable convolutions, where spatial filtering (depthwise convolution) and channel-wise feature combination (pointwise 1×1 convolution) are performed separately, significantly reducing computational complexity while improving feature learning.

The middle flow consists of eight repeated blocks of depthwise separable convolutions with residual connections, enabling deeper feature refinement and efficient gradient propagation. The exit flow continues with depthwise separable convolutions and transitions the network toward classification. A global average pooling layer is applied to reduce feature maps, followed by a fully connected layer and a SoftMax activation function to produce the final class predictions.

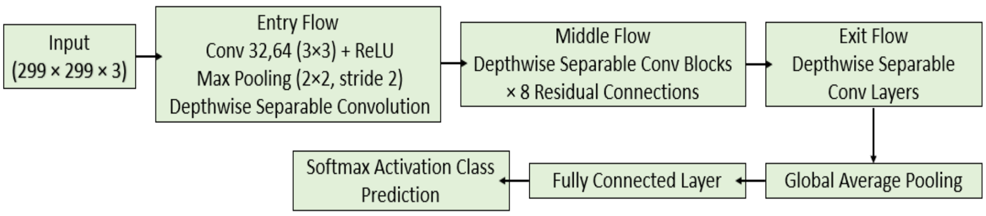

Fig. 5 Xception Model

Inception-V4

Szegedy *et al.*[26] introduced an advanced CNN architecture (Fig. 6) that is based on network-in-network design with the aim of improving model depth without incurring high computational expense. The network takes an RGB image of size 299×299 as input and starts with a Stem module for initial feature extraction using convolution and pooling layers. A multi-scale feature learning block follows in the form of four Inception-A blocks and then a block called Reduction-A is proposed to decrease the spatial dimension and increase the feature depth.

The architecture then uses seven Inception-B blocks and a Reduction-B block to reduce the dimensionality further. After that, three blocks of Inception are used to enhance the extracted features using lower spatial resolution. An average pooling layer is used to gather the spatial information into channel-wise feature vectors, which is followed by a dropout layer (rate = 0.8) to avoid overfitting. The network ends in a SoftMax layer which yields the probabilities for each class.

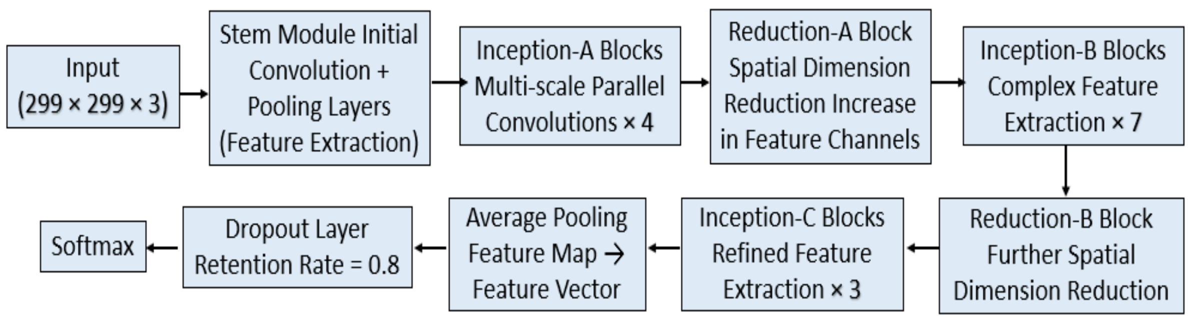

Fig. 6. Inception-v4 Architecture

Image preprocessing and Augmentation

The image data set was then preprocessed before training the models and then organized and augmented using the flow_from_directory method from TensorFlow's Image Data Generator to increase data diversity and help the model generalize. The images were then resized to the input dimensions of VGG-16 and ResNet-50 (224x224) and Xception and Inception-V4 (299x299) to match the selected deep learning architectures. Labels for each class were recoded into categorical (one-hot encoded) format, making them appropriate for the multi-class classification of four target classes. The data were partitioned before data augmentation to prevent information leakage and to provide unbiased model assessments. The initial image data set was split into three parts for training, validation and testing and augmentation was only performed on the training set. As such, no image augmentations were present in the validation and test sets to avoid a data leakage scenario and to provide a true test of the model's ability to generalize to unseen data. The dataset was split into a training set of 7,000 images, a validation set of 1,000 images and a test set of 2,000 images (70% / 10% / 20%). Multiplication of training data was performed with several data augmentation methods. The techniques used were horizontal flipping, zooming, shearing, rotation, shift and fill mode operations as explained in Table 2 and demonstrated in Fig. 7.

Table 1: Augmentation methods applied for dataset enhancement

Augmentation Technique	Values	Purpose
Horizontal Flipping	0.5	Improves invariance to left–right orientation variations
Zoom Range	0.3	Simulates variations in camera distance and scale
Shear Range	0.3	Introduces geometric distortions to enhance feature learning
Rotation Range	0.4	Accounts for rotational variations in leaf orientation
Width Shift Range	0.3	Handles horizontal displacement of leaf positions
Height Shift Range	0.3	Handles vertical displacement of leaf positions
Fill Mode	Nearest	Fills empty regions using adjacent pixel values to preserve visual continuity

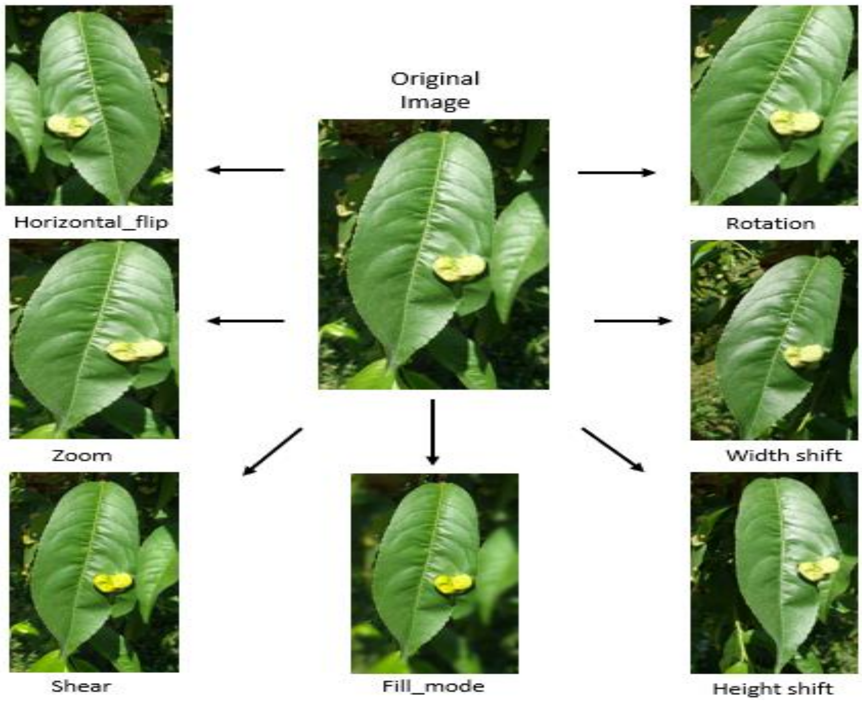

Fig. 7. Effects of data augmentation techniques applied to the dataset.

Implementation Setup and Training

All the experiments are carried out on a high performance computing setup to facilitate the efficient training of four different models namely VGG-16, ResNet-50, Xception and Inception-V4. The implementation is done on Ubuntu Linux OS with Python 3.10 and the TensorFlow deep learning framework (version 2.16.1) on JupyterLab. NVIDIA's DGX A100 GPU delivers 320 GB of GPU memory, 1 TB of system RAM and 15 TB of storage, significantly speeding up model training. This computation system has been used to train on a large scale maintaining stability and reducing training time. To make a fair comparative assessment between the architectures, the same data partitions, data augmentation, optimizers, training epochs and batch size were used for all. The hyperparameters for all models were fixed to reduce experimental bias and facilitate comparison of model architecture.Hyperparameters such as loss function, batch size and learning rate were standardized for all models to reduce experimental bias and facilitate comparison of model architecture. While the other architectures used transfer learning, equivalent training conditions were used to ensure an unbiased evaluation of the learning capabilities of the architectures on the same sets of data acquired under field conditions.

To ensure the experimental repeatability and stability, the same computational conditions, fixed dataset partitions and standardized preprocessing steps were used in model training. The Adam optimizer was used, with default values of momentum parameters (w1 = 0.9, w2 = 0.999), learning rate = 0.0001 and batch size = 32. Categorical cross-entropy loss was used for optimization due to its suitability for multi-class classification. Convergence behavior and generalization performance was checked with the validation set during training.

The pre-trained weights for the transfer learning-based architectures such as VGG16, ResNet50 and Xception were obtained from the ImageNet dataset to use the previously learned feature representations. The convolutional base of each architecture was first frozen to maintain the generalised feature extraction ability learnt from pre-training. Later, the existing classification levels were replaced by fully connected layers for peach leaf disease classes and the new proposed set was trained in the newly added classifier layers. This approach to transfer learning boosted the efficiency of training, speeded up the convergence and improved the generalization of the model on the field-acquired dataset.

Table 2: Computational Efficiency of Models

Model	Depth (Layers)	Parameters (Millions)	Input Size	Total Training Time
VGG16	16	138.4M	224×224×3	18 hours
Xception	71	22.9M	299×299×3	14 hours
ResNet50	50	25.6M	224×224×3	11 hours
InceptionV4	164	42.7M	299×299×3	38 hours

To get a good understanding of how well the model performs, multiple metrics were used. Apart from classification accuracy, precision, recall and F1-score were also calculated to evaluate the model's behavior for each class, especially the reliability of prediction and discrimination among the classes. These metrics can give a comprehensive view of the performance and reliability of the trained models.

Confusion Matrix: A confusion matrix is used to assess the model of classification, with the actual classes on the x-axis and predicted classes on the y-axis.

Precision

Precision, or the positive predictive value, is the percentage of positive examples that are correctly classified as positive. It reflects the percentage of times the model's predictions are correct. It can be determined by:

$$\text{Precision} = \frac{TP}{TP+FP}$$

Recall

Recall, also known as the true positive rate or sensitivity, is the percentage of true-positive data points that are classified as such by the model. It demonstrates the model's accuracy in identifying all pertinent cases. It is defined as:

$$\text{Recall} = \frac{TP}{TP+FN}$$

F1 Score

The F1 score is a combination of both precision and recall and is considered as a harmonic mean of both. It is between 0 and 1; 1 means it is working 100% of the time. It is determined by:

$$F1 = \frac{2\times [\text{Precision} \times \text{Recall}]}{\text{Precision} + \text{Recall}}$$

Accuracy

The accuracy is the percentage of correct classifications (positive and negative cases) to the total number of instances. It is given by:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \times 100$$

Results

Classification performance

The four convolutional neural network (CNN) architectures VGG-16, ResNet-50, Xception and Inception-V4 have been tested in terms of training and validation loss (shown in Fig. 8(a)-(d)), classification accuracy (shown in Fig. 9(a)-(d)) and confusion matrix based metrics. The study of training and validation loss gives useful insights to the convergence behaviour and the generalization ability of the models of training. The loss during training for VGG-16 also decreased gradually to around 0.19, while the loss during validation stabilized at 0.25-0.30, suggesting a significant difference. By contrast, ResNet-50 showed better convergence, training loss dropping to about 0.06 and validation loss stabilising at about 0.20-0.25, resulting in improved generalisation and less overfitting. For the Xception model, the training loss was more or less around 0.53 and the validation loss ranged between 0.66 and 0.72 during training. An instability suggests poor learning effectiveness and limited generalization ability.

On the other hand, Inception-V4 exhibited the best convergence, having the training loss converge to about 0.01 and the validation loss to about 0.10, with little gap between the two. This pattern is an example of good generalization and/or feature representation learning.

Inception-V4 attained the highest accuracy of 97.50%, followed by ResNet-50 with 94.49%, VGG-16 with 92.04% and Xception with 75.95%. The superior performance of Inception-V4 can be explained by its capability of efficient extraction of multi-scale features, whereas the lower accuracy of Xception is because it is not able to learn the discriminative features under the provided training settings.

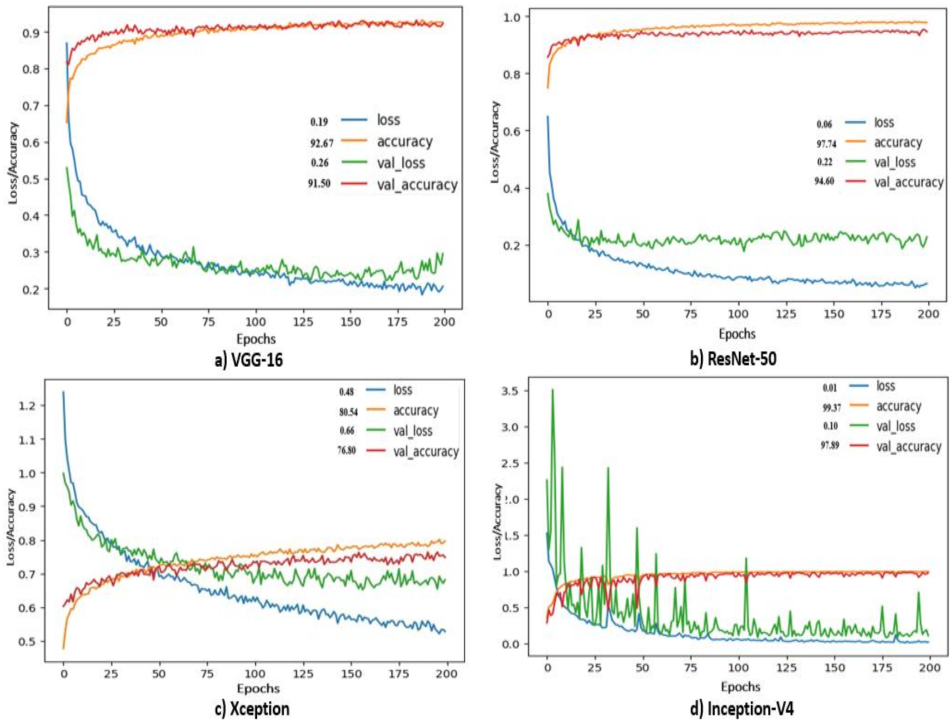

Fig. 8 Accuracy/Loss curves of a) VGG-16 b) ResNet-50 c) Xception d) Inception-V4

Confusion matrices (Fig. 9(a)–(d)) were used to further assess the results with a detailed analysis of the true positives (TP), false positives (FP), false negative (FN) and true negative (TN) for the four classes: leaf curl, rust, shot hole and healthy leaves. The TP values were 440, 446, 500 and 455 for leaf curl, rust, shot hole and healthy leaves, respectively for VGG-16. However, it had relatively high FN values meaning that some diseased and healthy samples were categorized into other categories. The high values for FP also caused some loss in precision. On the other hand, ResNet-50 performed better with TP values of 431, 482, 504 and 473 for each class respectively. It had a significant decrease in FN and FP results than VGG-16, with higher sensitivity and precision. The high TN values also showed high specificity, which meant that the overall classification accuracy and class discrimination were good. The TP values of Xception were comparatively low, namely 344, 337, 426 and 394. It also exhibited high FN and FP proportions by class, suggesting more misclassifications and reduced capabilities to separate the categories of disease. Its low TN values also emphasized the fact that it was not discriminative. On the other hand, the best performance was obtained by Inception-V4 with the highest TP values 475, 471, 501 and 503 for the four classes. The model had very low FN and FP values and high TN values. The near diagonal dominance in its confusion matrix indicates highly accurate and consistent classification with very low misclassification rates across all categories.

Models exhibiting lower validation loss and minimal divergence from training loss, such as Inception-V4 and ResNet-50, achieved higher TP rates and lower FP and FN, leading to superior performance. Conversely, Xception with a higher validation loss demonstrated reduced classification accuracy due to increased misclassification.

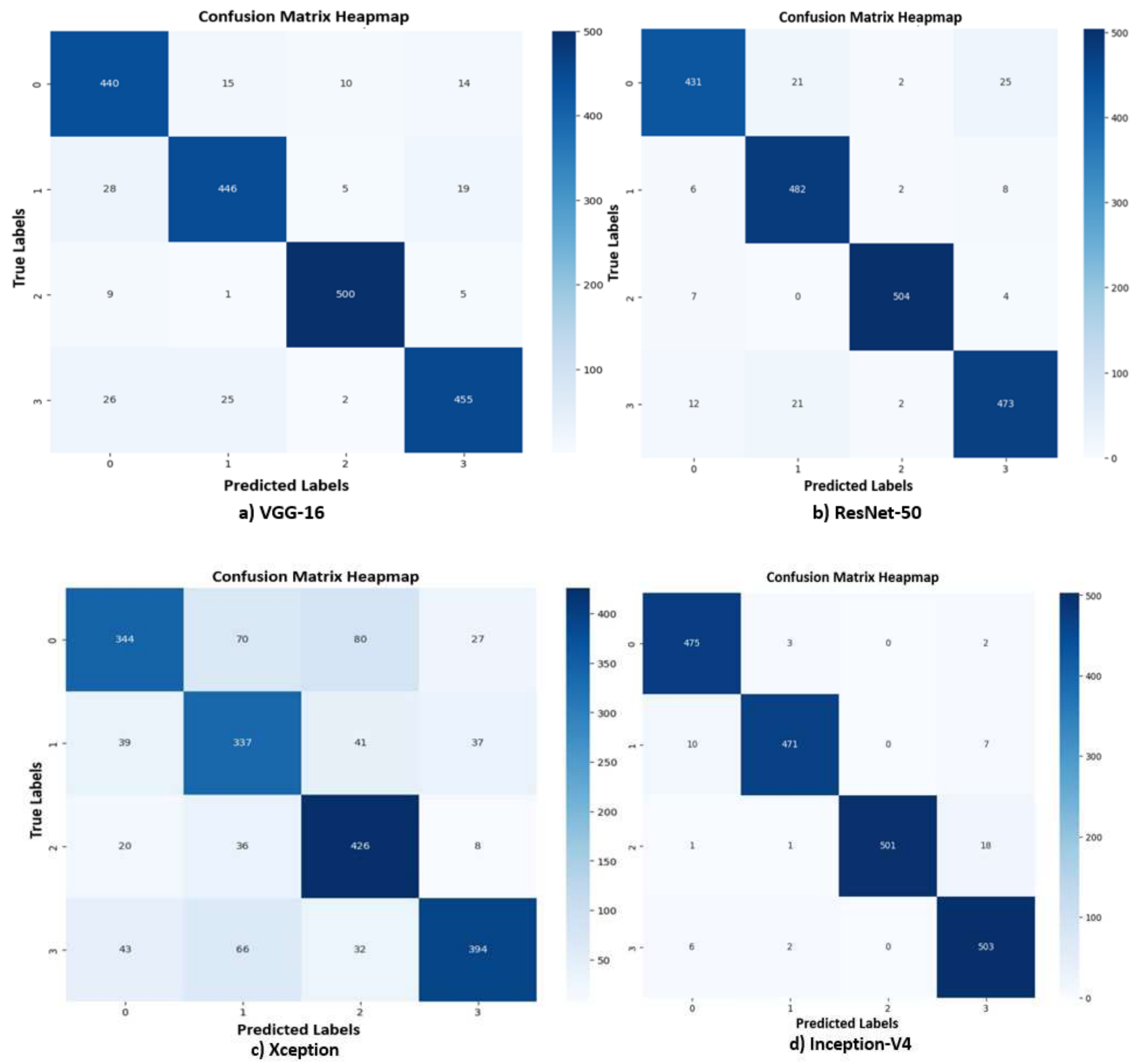

Fig. 9 Multi-Model Confusion Matrix Comparison of a) VGG-16 b) ResNet-50 c) Xception d) Inception-V4

Table 3: **Classification Report of the four Convolutional Neural Network Models**

Model	Class	Precision (%)	Recall (%)	F1-score (%)	Support	Accuracy (%)
VGG-16	Healthy	87	92	90	479	**92**
	Leaf Curl	92	90	91	498	
	Rust	97	97	97	515	
	Shot Hole	92	90	91	508	
	Macro Average	**92**	**92**	**92**	**2000**	
	Healthy	95	90	92	479	

ResNet-50	Leaf Curl	92	97	94	498	**94**
	Rust	99	98	98	515	
	Shot Hole	93	93	93	508	
	Macro Average	**94**	**94**	**94**	**2000**	
Xception	Healthy	77	66	71	521	**75**
	Leaf Curl	66	74	70	454	
	Rust	74	87	80	490	
	Shot Hole	85	74	79	535	
	Macro Average	**75**	**75**	**75**	**2000**	
Inception-V4	Healthy	97	99	98	480	**97**
	Leaf Curl	99	97	98	488	
	Rust	100	96	98	521	
	Shot Hole	95	98	97	511	
	Macro Average	**98**	**98**	**98**	**2000**	

The classification performance of the four models on the test dataset shows differences in their ability to accurately distinguish between peach leaf disease classes (Table 3). Inception-V4 achieved the highest overall performance, with a macro-average precision, recall and F1-score of 98% and an overall accuracy of 97%, demonstrating consistently high scores across all classes, including near-perfect precision for the Rust class at 100%. This suggests its good performance on discriminative and multi-scale feature capture. ResNet-50 had a macro-average value of 94% and an overall accuracy of 94%, with good performance in all classes, especially in the Rust class, with an F1 score of 98%, demonstrating a balanced and reliable performance and the effectiveness of residual learning in improving feature representation. VGG-16 performed somewhat worse but still acceptable with a macro-average accuracy of 92% and an overall accuracy of 92%, showing high accuracy on Rust class (97%) and a moderate accuracy on Healthy class (87%), suggesting that the model might be struggling to classify leaves as healthy or diseased. Unlike Xception, the performance of the latter was low as reflected in its macro-average of 75% and overall accuracy of 75%, with lower precision and recall seen across most classes, especially Healthy (F1 = 71%) and Leaf Curl (F1 = 70%), indicating a lack of the ability to capture the subtle inter-class variations.

Performance comparison of CNN models

The training/learning and testing/accuracy of the 4 models are summarized in Table 4 and shown graphically in Fig. 10 as accuracy curves over 200 epochs. The best and most stable convergence was demonstrated in InceptionV4 followed by ResNet50. VGG16 had moderate stability with small fluctuations and Xception had slow learning and more variations.

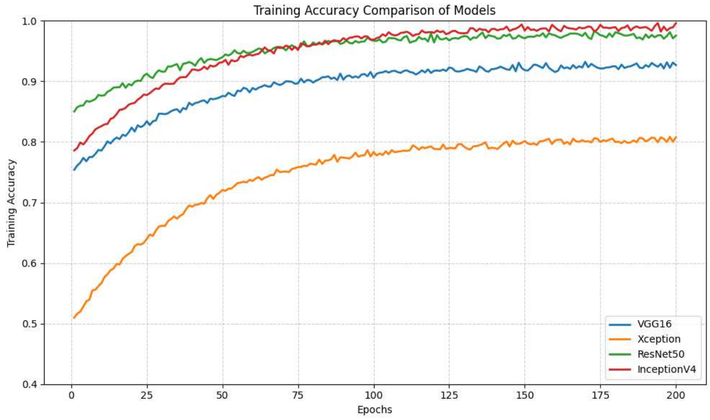

Fig 10: Training accuracy comparison of VGG16, ResNet50, Xception and InceptionV4 over 200 epochs

Table 4: Performance Comparison of Models

Model	Training Accuracy (%)	Testing Accuracy (%)
VGG16	92.67	92.04
Xception	80.54	75.95
ResNet50	97.74	94.49
InceptionV4	99.11	97.50

Discussion

The experimental results demonstrate the potential of DCNNs in detecting and classifying peach leaf diseases automatically. Lighting, background complexity, leaf orientation and natural noise variations were included in the data set used in this study under real field conditions. This variability also makes the classification task more difficult but ensures that the models learned are stronger and can be used in real agricultural situations. The classification accuracy obtained on the four architectures evaluated VGG16, ResNet50, Xception and InceptionV4, was highest for InceptionV4, followed by ResNet50 and VGG16 and lowest for Xception. Such performance variation is due to factors such as architectural design and training strategy, which determines the model's performance in capturing complex disease patterns[27,28].

This is likely because Inception-V4 have improved capabilities in multi scale feature extraction that helps them to identify disease symptoms in different spatial resolution[29]. Peach leaf diseases like rust, shot hole and leaf curl show unique symptoms from the leaf, including lesions and perforations, to the leaf as a whole, including morphological distortions and discoloration patterns. This inception modules process operates concurrently on several receptive fields, enabling the network to detect subtle textural abnormalities and more general structural changes. In contrast, ResNet50 performed well due to residual learning, but it performed comparatively lower, suggesting that it may not be as effective in capturing subtle symptom heterogeneity that may exist under natural field conditions[30,31]. The accuracy of VGG16 on peach leaf disease is 92%. The convolutional deep network was deep and uniform, effectively extracting disease discriminative information, thus having good classification performance. But because of many parameters, the computational cost and efficiency were more than the architecture of InceptionV4 and ResNet50[32]. Xception, on the other hand, is a comparatively weaker performance indicating that the depthwise separable convolutions might not have been as effective in complex disease-specific pattern learning in heterogeneous orchard environments[33,34, 35].

Class-wise analysis of prediction shows that all the models were performing well in the classification of healthy leaves, leaf curl, rust and shothole diseases with clearly visible symptoms. In general, the number of misclassifications was small, however and mostly occurred in the Xception model. This means that the other architectures had more stable prediction performance across classes with the same training conditions, whereas the prediction performance of the Xception was relatively unreliable. One of the main results of this study is that the training strategy plays a significant role in the performance of the model. Whereas transfer learning helped some models[36] to converge quicker, the better performance of InceptionV4 trained from scratch clearly shows that learning the domain-specific features directly could yield better classification accuracy if the required amount of data and training time is available. The results make it clear that architectural design and feature extraction is critical to ensure reliable and accurate detection and classification of disease and Inception-V4 is the most appropriate model for the peach disease detection and classification[37].

While the proposed framework showed good classification performance, there are some limitations to be noted. The images used for training and testing the models were mostly from orchards in a specific geographical region of Kashmir and may not have higher generalizability under different environmental and epidemiological settings. Model robustness may be affected by differences in climatic conditions, cultivar susceptibility, severity of disease and orchard management. Future research is needed to assess the model adaptability and deployment potential by using geographically diverse data sets and external validation across a variety of orchard systems.

Conclusion

The study aimed to design and test a deep learning-based approach for automatic detection and classification of peach leaf diseases in the actual field condition with a large scale image dataset from temperate orchards of Kashmir. All the models learned disease-associated visual patterns and the results showed that the models performed differently when evaluated together in comparison to VGG-16, ResNet-50, Xception and Inception-V4. The results showed that Inception-V4 has the best classification accuracy, meaning it has the best performance for the extraction of multi-scale and disease-specific features directly from field-acquired images. The results highlight the relevance of the architecture selection and training strategy of image analysis in agriculture and show that deep learning methods can be used in precision horticulture for reliable disease diagnosis. Future work should include an extension of the disease categories, introduction of external validation datasets and integration of the framework into mobile assisted diagnostic platforms for real-time monitoring disease in the orchard.

Data availability:

The dataset generated during the current study are available from the corresponding author on reasonable request.

Author contributions:

Mankotia, A., Rasool, S., Shah, H.N., Bhat, M.A**:** Wani, S.Q: **Conceptualization, Methodology, Data curation,** Mankotia, A., Rasool, S, Bhat, M.A, Nahida Chisti: **Investigation, Formal analysis, Software, Validation, Visualization,** Dar, M.S., Bhat, Z.A., Pandit, A.H., Kirmani, M.M., Wani, S.Q., Nisar, S.B., Farooq, Z Mujtaba Abdullah Kumar**: Writing original draft preparation**

Funding :

The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Competing interests:

The authors declare no competing interests.

Declaration of generative AI and AI-assisted technologies in the writing process:

During the preparation of this work, the authors used OpenAI exclusively to improve the clarity and readability of the language. The authors reviewed and edited the output as necessary and take full responsibility for the content of this publication.

References:

1. Cruz, A.C., Luvisi, A., De Bellis, L. and Ampatzidis, Y., 2017. X-FIDO: An effective application for detecting olive quick decline syndrome with deep learning and data fusion. *Frontiers in plant science*, *8*, p.1741.
2. Saleem, M. H., Potgieter, J., & Arif, K. M. (2022). A performance-optimized deep learning-based plant disease detection approach for horticultural crops of New Zealand. *IEEE Access*, *10*, 89798-89822.
3. Deng, L. and Yu, D. 2014. Deep learning: Methods and applications. *Foundations and Trends in Signal Processing* **7**(3-4): 197-387.
4. Iftikhar, M., Kandhro, I. A., Kausar, N., Kehar, A., Uddin, M., & Dandoush, A. (2024). Plant disease management: a fine-tuned enhanced CNN approach with mobile app integration for early detection and classification. *Artificial Intelligence Review*, *57*(7), 167.
5. Pydipati, R., Burks, T. F., & Lee, W. S. (2006). Identification of citrus disease using color texture features and discriminant analysis. *Computers and electronics in agriculture*, *52*(1-2), 49-59.
6. P. R. Rothe and R. V. Kshirsagar, "Cotton leaf disease identification using pattern recognition techniques," *2015 International Conference on Pervasive Computing (ICPC)*, Pune, India, 2015, pp. 1-6, doi: 10.1109/PERVASIVE.2015.7086983.

7. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems (pp. 1097–1105).
8. Mohanty SP, Hughes DP and Salathé M (2016) Using Deep Learning for Image-Based Plant Disease Detection. Front. Plant Sci. 7:1419. doi: 10.3389/fpls.2016.01419
9. Ferentinos, K. P. 2018. Deep learning models for plant disease detection and diagnosis. Computers and Electronics in Agriculture 145: 311-318.
10. Kawasaki, Y., Uga, H., Kagiwada, S. and Iyatomi, H. 2015. Basic study of automated diagnosis of viral plant diseases using convolutional neural networks. In: Advances in Visual Computing: Eleventh International Symposium. Las Vegas, NV, USA, 638-645, December.
11. Dos Santos Ferreira, A., Freitas, D. M., da Silva, G. G., Pistori, H. and Folhes, M. T. 2017. Weed detection in soybean crops using ConvNets. Computers and Electronics in Agriculture 143: 314-324.
12. Wang, G., Sun, Y. and Wang, J. 2017. Automatic image-based plant disease severity estimation using deep learning. Computational Intelligence and Neuroscience 1: 2917536.
13. Too, E. C., Yujian, L., Njuki, S. and Yingchun, L. 2019. A comparative study of fine-tuning deep learning models for plant disease identification. Computers and Electronics in Agriculture 161: 272-279.
14. Rozlan, S. and Hanafi, M. 2022. Efficacy of chili plant diseases classification using deep learning: a preliminary study. Computer Science 25: 1442-1449.
15. Kumar, A. and Kumar, A. 2023. Plant disease detection using VGG16. International Journal of Creative Research Thoughts 11(1): 770-775.
16. Howlader, M. R., Habiba, U., Faisal, R. H. and Rahman, M. M. 2019. Automatic recognition of guava leaf diseases using deep convolution neural network. In: International Conference on Electrical, Computer and Communication Engineering. Cox's Bazar, Bangladesh, 1-5, February.
17. Sutaji, D. and Yıldız, O. 2022. LEMOXINET: Lite ensemble MobileNetV2 and Xception models to predict plant disease. Ecological Informatics 70: 101698.
18. Alimboyong, C. R., & Hernandez, A. A. (2019). *An improved deep neural network for classification of plant seedling images.* In 2019 IEEE 15th International Colloquium on Signal Processing & Its Applications (CSPA) (pp. 217–222), Penang, Malaysia. IEEE. https://doi.org/10.1109/CSPA.2019.8696009
19. Hasan, M. J., Mahbub, S., Alom, M. S. and Nasim, M. A. 2019. Rice disease identification and classification by integrating support vector machine with deep convolutional neural network. In: First International Conference on Advances in Science, Engineering and Robotics Technology. Dhaka, Bangladesh, 1-6, May.
20. Pscheidt JW. Leaf curl. In: Ogawa JM, Zehr EI, Bird GW, Ritchie DF, Uriu K, Uyemoto JK, editors. Compendium of stone fruit diseases. St. Paul, MN: APS Press; 1995. p. 19–20.
21. Adaskaveg, J. E., Ogawa, J. M., & Butler, E. E. (1990). Morphology and ontogeny of conidia in *Wilsonomyces carpophilus*, gen. nov. and comb. nov., causal pathogen of shot hole disease of Prunus species. Mycotaxon, 37, 275–290.
22. Shin, H. D., Lee, S. H., Jeon, C. H., & Choi, W. I. (2019). First report of rust caused by Tranzschelia discolor on peach (Prunus persica) in Korea. *Plant Disease*, *103*(4), 764.
23. Simonyan, K. and Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. In: Proceedings of the International Conference on Learning Representations. arXiv preprint arXiv:1409.1556.
24. He, K., Zhang, X., Ren, S. and Sun, J. 2016. Deep residual learning for image recognition. In: Proceedings of the Institute of Electrical and Electronics Engineers Conference on Computer Vision and Pattern Recognition pp. 770-778.

25. Chollet, F. 2017. Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the Institute of Electrical and Electronics Engineers Conference on Computer Vision and Pattern Recognition pp. 1251-1258.
26. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. 2017. Inception-v4, inception-ResNet and the impact of residual connections on learning. In: Proceedings of the Thirty-first Association for the Advancement of Artificial Intelligence Conference on Artificial Intelligence. San Francisco, CA, USA, 4278–4284, February.
27. Zhang, D., Yang, H., Cao, J., Gong, Q., & Kumaran, S. R. (2023). Peach leaf diseases identification using convolutional neural network and Fastai framework.
28. Chen, K., et al. (2024). Integration of Image and Sensor Data for Improved Peach Tree Disease Detection. *Agriculture*, 14(6), 797.
29. Sharma, V., Tripathi, A. K., & Mittal, H. (2023). DLMC-Net: Deeper lightweight multi-class classification model for plant leaf disease detection. *Ecological informatics*, *75*, 102025.
30. Srijah, S., Sridevi, S., Ezhil, G.R., Saranya, T. (2026). Tomato Leaf Disease Classification Using an Improved ResNet50 Architecture. In: Das, S., Paprzycki, M., Ghosh, A., Bianchini, M. (eds) Advanced Computing and Intelligent Technologies. ICACIT 2024. Lecture Notes in Networks and Systems, vol 1359. Springer, Singapore. https://doi.org/10.1007/978-981-96-4933-4_36
31. Mukti, I. Z. and Biswas, D. 2019. Transfer learning based plant diseases detection using ResNet50. In: Fourth International Conference on Electrical Information and Communication Technology. Khulna, Bangladesh, 1-6, December.
32. Rustam, Z., Sari, R., Asbudi, H. L., & Susilawati, F. E. (2026). Rice leaf disease classification using ResNet-50. *Brilliance: Research of Artificial Intelligence*, *6*(1).
33. Hukkeri, G. S., Soundarya, B. C., Gururaj, H. L., & Ravi, V. (2024). Classification of various plant leaf disease using pretrained convolutional neural network on imagenet. *The Open Agriculture Journal*, *18*(1).
34. Kamal, K.; Yin, Z.; Wu, M.; Wu, Z. 2019. Depthwise separable convolution architectures for plant disease classification. *Comput. Electron. Agric.165*: 104948.
35. Pacal, I., Kunduracioglu, I., Alma, M. H., Deveci, M., Kadry, S., Nedoma, J., ... & Martinek, R. (2024). A systematic review of deep learning techniques for plant diseases. *Artificial Intelligence Review*, *57*(11), 304.
36. Xu, M., Yoon, S., Jeong, Y., & Park, D. S. (2022). Transfer learning for versatile plant disease recognition with limited data. *Frontiers in plant science*, *13*, 1010981.
37. Ashurov, A. Y., Al-Gaashani, M. S., Samee, N. A., Alkanhel, R., Atteia, G., Abdallah, H. A., & Saleh Ali Muthanna, M. (2025). Enhancing plant disease detection through deep learning: a Depthwise CNN with squeeze and excitation integration and residual skip connections. *Frontiers in Plant Science*, *15*, 1505857.