MACHINE LEARNING

Plant Leaf Disease Classification Using Deep Learning with ResNet-50

Guntha Sujith, Dodiyam Sandula Guru Sudheer, Saraka Varun Simha, Mallekonda Harsha Vardhan, R. Pavan Kumar

Department of Electronics and Communication Engineering, Rajiv Gandhi University of Knowledge Technologies (RGUKT), RK Valley, Andhra Pradesh, India

Published November 2026

Abstract

Background/Objective: Plant diseases can reduce crop yield and quality. Identifying visible symptoms at an early stage may help farmers respond before damage becomes widespread. This study evaluates an image-level classifier that assigns leaf photographs to 38 healthy and diseased categories using ResNet-50.

Methods: The New Plant Diseases Dataset (Augmented) contains 70,295 training images and 17,572 validation images. Images were resized to 224 × 224 pixels and normalised. Random horizontal flipping, rotation and colour jitter were applied during training. A ResNet-50 model with a 38-class output layer was trained from random initialisation for three epochs on a CPU, using cross-entropy loss and the AdamW optimiser. Pretrained ImageNet weights could not be downloaded in the experimental environment.

Results: The best checkpoint, obtained at epoch 3, achieved 99.80% validation accuracy: 17,536 of 17,572 images were classified correctly, while 36 were misclassified. Among the reported per-class results, the lowest recall was 0.96 for corn Cercospora/Gray leaf spot, and the lowest precision was 0.97 for corn Northern Leaf Blight.

Conclusion: The model achieved high accuracy on the dataset's validation split. However, this split is not an independent test set, and the dataset contains augmented images. The result should therefore be interpreted as a baseline under controlled conditions; evaluation on independent field images is still required.

Keywords: convolutional neural network; deep learning; leaf images; plant disease classification; ResNet-50

1 Introduction

Crop health is closely linked to food production and agricultural productivity. When a plant disease is not identified early, it can spread and affect both crop yield and quality. In many cases, the first visible signs appear on leaves as spots, discolouration, lesions, mould-like growth or changes in texture. Recognising these symptoms from photographs, however, can be difficult. Their appearance may vary with the crop, disease stage and image conditions, and accurate interpretation often requires experience.

Deep learning provides one way to support the analysis of leaf images. A convolutional neural network (CNN) learns visual features from labelled examples and uses them to predict a class for a new image. This study addresses image-level classification: the input is a complete leaf photograph, and the output is one of 38 class labels. The proposed model does not identify the location of lesions,

generate segmentation masks or estimate disease severity. Those tasks require spatial annotations and are outside the scope of the present work.

The main contributions of this study are as follows:

- A training and evaluation workflow for classifying plant leaf images into 38 categories using ResNet-50.
- An evaluation that reports class-wise precision, recall and F1-score, together with a confusion matrix, rather than relying on accuracy alone.
- A description of the experimental constraints, including training from random initialisation, a three-epoch schedule and evaluation on an augmented dataset.

1.1 Related work and design context

Earlier approaches to plant disease recognition often used manually designed features, including colour histograms, texture descriptors and shape measurements. These features were then supplied to classifiers or used in image-processing methods such as thresholding and segmentation. Such approaches can be effective when symptoms are visually consistent, but their performance may be affected by changes in illumination, camera settings, background clutter and leaf age. Methods such as support vector machines and k-nearest neighbours also depend strongly on the quality of the features provided to them.

Convolutional neural networks reduce the need to design features manually by learning representations directly from image pixels. Early layers generally capture simple visual patterns, such as edges and colour transitions, while deeper layers can represent more complex textures and shapes. Mohanty, Hughes and Salathé demonstrated the potential of deep learning for plant disease identification using the PlantVillage dataset [2]. Too et al. compared fine-tuned deep-learning architectures for the same general task [3], and Abade et al. presented a systematic review of CNN-based plant disease recognition [4].

Training very deep neural networks can be difficult because optimisation and gradient propagation become more challenging as network depth increases. He et al. introduced residual learning to address this issue [1]. In a residual block, a learned function $F(x)$ is added to the block input through an identity shortcut, giving $y = F(x) + x$ when the dimensions are compatible. This structure provides a direct path for information and gradients through the network. ResNet-50 uses bottleneck blocks containing 1×1, 3×3 and 1×1 convolutions. Its four stages contain 3, 4, 6 and 3 blocks, respectively; the 1×1 convolutions adjust channel dimensions, while the 3×3 convolution performs spatial feature extraction [6].

The focus of this work is the implementation and evaluation of a 38-class classification pipeline. The pipeline uses consistent image preprocessing, model checkpointing and class-wise evaluation. The experimental results are reported alongside the limitations of the dataset and training setup.

2 Methods

2.1 Dataset

The experiments use the New Plant Diseases Dataset (Augmented), obtained from Kaggle [5]. The images are organised into class-specific folders within separate training and validation directories; each folder name is used as the corresponding class label. The 38 categories include healthy and diseased

leaves from apple, blueberry, cherry, corn, grape, orange, peach, pepper, potato, raspberry, soybean, squash, strawberry and tomato (Table 1).

Table 1. Dataset split.

Subset	Images	Purpose
Training	70,295	Learning model parameters
Validation	17,572	Model selection and monitoring
Total	87,867	All inspected images

All 38 class folders were present in both directories, and the class-to-index mapping was checked for consistency between the training and validation sets. The classes are approximately balanced, with around 1,600–2,000 training images and 410–505 validation images per class. The dataset does not provide a separate test directory; accordingly, all results in this paper are validation results.

2.2 Preprocessing and augmentation

Each image was resized to 224 × 224 pixels and converted into a floating-point tensor. The image channels were normalised using the mean [0.485, 0.456, 0.406] and standard deviation [0.229, 0.224, 0.225]. During training, augmentation consisted of random horizontal flipping with a probability of 0.5, random rotation of up to 15 degrees, and colour jitter with brightness, contrast and saturation values set to 0.2. Validation images were resized, converted to tensors and normalised, but were not augmented. This kept the validation preprocessing consistent across evaluation runs.

2.3 Network architecture

The model uses the torchvision implementation of ResNet-50 [6]. Following the final pooling stage, the network produces a 2,048-dimensional feature vector. The original fully connected layer, which outputs 1,000 ImageNet class scores, was replaced with a new fully connected layer producing scores for the 38 leaf categories.

The model was initialised with weights=None. An attempt was made to load the default ImageNet weights, but the download was unsuccessful because the runtime environment could not connect to the host. The experiments therefore used random initialisation and trained the network from scratch. The model was not fine-tuned from pretrained ImageNet weights, and the procedure should not be described as transfer learning.

2.4 Classification and loss

Let $z \in \mathbb{R}^{38}$ denote the output logits from the final fully connected layer. During training, cross-entropy loss is calculated by comparing these logits with the integer class labels. During inference, the logits can be converted by softmax into a probability distribution over the 38 classes. The class with the highest predicted score is selected as the output label.

2.5 Training configuration

The training settings are summarised in Table 2. ReduceLROnPlateau monitors the validation loss and is configured to reduce the learning rate by a factor of 0.5 when the monitored loss stops improving, subject to the specified patience. The latest model checkpoint is saved after each epoch. A separate best-model checkpoint is updated whenever validation accuracy improves.

Table 2. Training configuration.

Parameter	Value
Model	ResNet-50 (torchvision)
Initialisation	Random (pretrained weights unavailable)
Input size	224 × 224 × 3
Output classes	38
Batch size	32
Epochs	3
Loss	CrossEntropyLoss
Optimiser	AdamW
Initial learning rate	0.001
Weight decay	0.0001
Scheduler	ReduceLROnPlateau (factor 0.5, patience 2)
Device	CPU

2.6 Evaluation metrics

Accuracy is calculated as the proportion of validation images for which the predicted class matches the ground-truth label. For each class, precision is TP / (TP + FP), recall is TP / (TP + FN), and F1-score is the harmonic mean of precision and recall, 2PR / (P + R). Here, TP, FP and FN denote true positives, false positives and false negatives, respectively. A 38 × 38 confusion matrix is used to examine the class labels that are most often confused. Detection mean average precision (mAP) and segmentation mean intersection over union (mIoU) are not reported because the proposed model performs image classification and does not predict bounding boxes or segmentation masks.

2.7 Classification pipeline

Figure 1 summarises the classification pipeline. An input leaf image is first preprocessed and then passed through the ResNet-50 feature extractor. The resulting feature representation is processed by the 38-class fully connected layer, and the class with the highest score is returned as the predicted label.

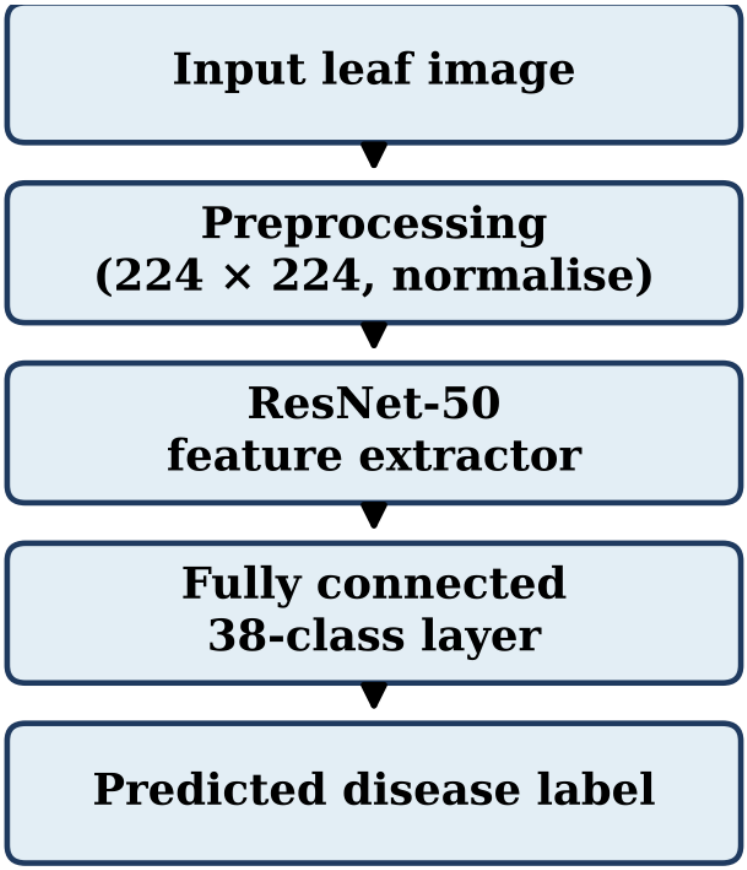


Figure 1. Classification pipeline of the proposed ResNet-50 model.

3 Results

3.1 Training behaviour

The experiments were implemented in PyTorch and executed on a CPU. The training and validation results for the three epochs are reported in Table 3 and illustrated in Figure 2.

Table 3. Epoch-wise training and validation results.

Epoch	Train loss	Train acc.	Val. loss	Val. acc.
1	0.0406	98.78%	0.0145	99.57%
2	0.0254	99.23%	0.0141	99.53%
3	0.0186	99.42%	0.0066	99.80%

Across the three epochs, training loss decreased from 0.0406 to 0.0186, while training accuracy increased from 98.78% to 99.42%. Validation accuracy changed from 99.57% in the first epoch to 99.53% in the second, before increasing to 99.80% in the third epoch. The validation loss at epoch 3 was 0.0066. Validation accuracy being higher than training accuracy can occur in this setup because the training images are randomly augmented, whereas the validation images are not. Consequently, the model is evaluated on unaugmented images but trained on images that have undergone additional transformations.

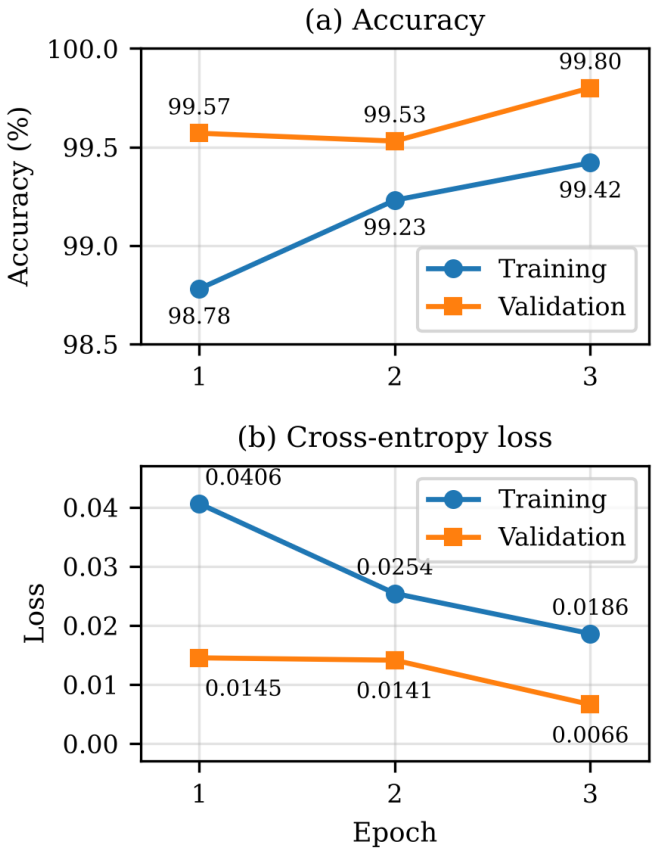


Figure 2. Training and validation (a) accuracy and (b) loss over three epochs.

3.2 Overall performance

The checkpoint selected at epoch 3 correctly classified 17,536 of the 17,572 validation images. The remaining 36 images were misclassified. The resulting validation accuracy was 99.7951%, which is 99.80% when rounded to two decimal places.

3.3 Per-class performance

After rounding to two decimal places, most classes had precision, recall and F1-scores of 1.00. The classes with one or more reported values below 1.00 are listed in Table 4.

Table 4. Classes with rounded metrics below 1.00.

Class	Precision	Recall	F1	Support
Corn – Cercospora / Gray leaf spot	1.00	0.96	0.98	410
Corn – Northern Leaf Blight	0.97	1.00	0.98	477
Tomato – Late blight	1.00	0.99	0.99	463
Tomato – Septoria leaf spot	1.00	0.99	1.00	436
Tomato – Target Spot	0.99	0.99	0.99	457
Tomato – Bacterial spot	0.99	1.00	1.00	425
Tomato – healthy	0.99	1.00	1.00	481

Among the classes listed in Table 4, corn Cercospora/Gray leaf spot had the lowest recall (0.96), while corn Northern Leaf Blight had the lowest precision (0.97). This pattern may indicate that some images from these two classes were confused with one another. However, the raw confusion-matrix counts were not available for confirmation, so this interpretation should be treated as a possibility rather than a confirmed result. The confusion matrix and the complete classification report were saved as CSV files. The full class-wise results can be obtained from the authors.

4 Discussion

4.1 Interpretation and limitations

The model achieved high classification accuracy across the 38 categories in the validation split after three epochs of training from random initialisation. This result indicates that the model learned to distinguish the classes represented in this dataset. However, the score must be interpreted in the context of the dataset composition and the experimental procedure. The limitations are described below.

1. **No independent test set:** The dataset contains training and validation directories but no separate test set. The reported values are therefore validation metrics. In addition, validation accuracy was used to select the best checkpoint, so it should not be interpreted as an unbiased estimate of performance on unseen data.
2. **Augmented dataset:** The dataset is an augmented collection of images. If images derived from the same original leaf are present in both the training and validation sets, similarity between those images could increase the measured validation accuracy. The extent of any such overlap was not established in this experiment. A split based on original source images would help assess this risk.
3. **Limited training and no pretraining:** The network was trained for three epochs from random initialisation. This experiment does not establish the model's maximum achievable performance. Training for longer and using ImageNet-pretrained weights may produce different results and should be investigated in future experiments.
4. **Controlled image conditions:** The dataset images are relatively standardised. The model was not evaluated on field photographs containing variations in lighting, occlusion, background clutter, camera quality or leaf orientation.
5. **Image-level classification only:** The model assigns one class label to each input image. It does not locate diseased regions, segment leaves or estimate disease severity.

Taken together, these limitations mean that the model should be regarded as a research prototype. Independent testing, particularly with field photographs, is necessary before considering its use in agricultural settings.

4.2 Future work

The following extensions could be considered in future work:

- Use ImageNet-pretrained weights and compare fine-tuning with the current training-from-scratch approach.
- Extend the training schedule and investigate early stopping and systematic learning-rate selection.

- Evaluate the model on an independent test set and on field photographs collected under varied conditions.
- Use Grad-CAM or a similar visualisation method to examine which image regions influence the model's predictions.
- Analyse the confusion matrix in greater detail, particularly for the corn and tomato categories, and collect additional examples for classes with weaker results.
- For lesion localisation or leaf segmentation, collect images with verified bounding-box or pixel-level annotations and evaluate task-specific models separately.

This work is relevant to Sustainable Development Goal 2 (Zero Hunger) because image-based crop-health monitoring may contribute to earlier identification of plant diseases. Any practical use, however, would require validation under field conditions.

5 Conclusion

This study implemented a ResNet-50 model to classify leaf images into 38 healthy and diseased categories. The images were resized and normalised, with data augmentation applied during training. The model was trained from random initialisation for three epochs on a CPU and achieved 99.80% accuracy on the dataset's validation split, correctly classifying 17,536 of 17,572 images. A small number of corn and tomato categories had per-class metrics below 1.00 after rounding. These results provide a baseline for the selected dataset, but they do not establish performance on independent or field-collected images. Future work should therefore include testing on independent data, evaluating ImageNet-pretrained initialisation and assessing the model using photographs captured under real agricultural conditions.

Declarations

Funding and Conflict of Interest: This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors. The authors declare that there is no conflict of interest regarding the publication of this manuscript. All listed authors contributed to the experimental design, model training, analysis, and manuscript preparation, and approved the final version for submission.

Acknowledgment

The authors thank the Department of Electronics and Communication Engineering, Rajiv Gandhi University of Knowledge Technologies (RGUKT), RK Valley, Andhra Pradesh, India, for academic guidance and institutional support.

References

[1] 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 (CVPR)*; 2016. p. 770–778. https://doi.org/10.1109/CVPR.2016.90

[2] Mohanty SP, Hughes DP, Salathé M. Using deep learning for image-based plant disease detection. *Frontiers in Plant Science*. 2016;7:1419. https://doi.org/10.3389/fpls.2016.01419

[3] Too EC, Yujian L, Njuki S, Yingchun L. A comparative study of fine-tuning deep learning models for plant disease identification. *Computers and Electronics in Agriculture*. 2019;161:272–279. https://doi.org/10.1016/j.compag.2018.03.032

[4] Abade AS, Ferreira PA, de Barros Vidal F. Plant diseases recognition on images using convolutional neural networks: A systematic review. *Computers and Electronics in Agriculture*. 2021;185:106125. https://doi.org/10.1016/j.compag.2021.106125

[5] vipoooool. *New Plant Diseases Dataset (Augmented)* [Internet]. Kaggle; [cited 2026 Nov 16]. Available from: https://www.kaggle.com/datasets/vipoooool/new-plant-diseases-dataset

[6] PyTorch Contributors. torchvision.models.resnet50 [Internet]. Torchvision documentation; [cited 2026 Nov 16]. Available from: https://pytorch.org/vision/stable/models/generated/torchvision.models.resnet50.html