Plant Disease Detection Using Deep Learning and MobileNet

Ankur Paul[1] **and Sandeep Singh Sandha**[2]

[1]**Guru Nanak Dev Engineering College, Ludhiana**
[2]**Punjab AI Excellence**

ABSTRACT

This report details the development of an AI-based plant disease detection system as part of a one-month industrial training at Punjab AI Excellence. The primary objective of the project was to leverage modern machine learning and deep learning techniques—particularly Convolutional Neural Networks (CNNs) and transfer learning—for accurate identification of plant diseases across multiple crop species using leaf images.

The project began with foundational learning and hands-on experiments with basic neural networks, evolving to advanced image classification by fine-tuning pre-trained models such as EfficientNet-B0 and MobileNetV2. A balanced and augmented multi-crop dataset was curated and utilized for model training and rigorous validation.

The final MobileNetV2-based models demonstrated high accuracy on both individual crop datasets and a combined multi-class classification task, with validation accuracies exceeding 95% for most crops. Real-world testing confirmed robust performance, though some limitations were observed in the presence of complex backgrounds and multiple diseases within a single image.

This report provides insights into neural network architecture, training pipelines, data balancing, augmentation strategies, and detailed evaluation using confusion matrices and per-class metrics. The findings underscore the impact of transfer learning in developing resource-efficient, mobile-friendly AI solutions for plant health monitoring. Recommendations are also made for future work to improve generalization to diverse, real-world agricultural environments.

Keywords: Plant Disease Detection, Deep Learning, CNN, MobileNet, Image Classification, Transfer Learning, Tomato Leaf, PlantVillage Dataset

1 INTRODUCTION

1.1 Problem Statement

Plant diseases significantly impact crop productivity and food security across the globe. A recent comprehensive assessment by CropLife India, Krishak Jagat (2023), indicates that plant diseases and pests cause annual economic losses of Rs. 2 lakh crores (approximately $24 billion) to Indian agriculture. Early and accurate identification of these diseases is critical for timely intervention and effective management. Traditional methods rely heavily on expert evaluation and manual inspection, which are not always feasible, especially for small-scale farmers or those in rural areas.

The primary objective of this project is to leverage modern machine learning and deep learning techniques, particularly Convolutional Neural Networks (CNNs) and transfer learning, for accurate identification of plant diseases using leaf images. The system aims to develop an AI application using Foundation Models (Huyen (2024)) to automate the detection process using mobile-friendly AI models that can run on lightweight devices.

By building a robust classification model based on image data, the system seeks to offer real-time, scalable, and cost-effective support to farmers, enabling them to make informed decisions about crop treatment without needing expert knowledge or equipment. This addresses the real-world problem of limited access to reliable and timely plant disease diagnosis.

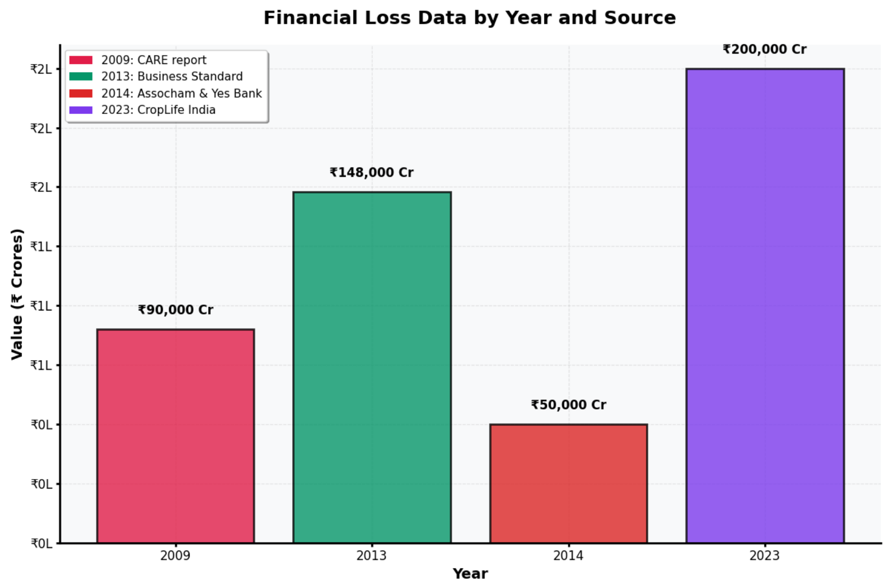


Figure 1. Estimated Financial Loss Data by Year and Source — Sources: Government of India (2009), Sud (2013), PTI (2014), Krishak Jagat (2023)

1.2 Objectives

The primary objective of this project is to leverage modern machine learning and deep learning techniques—particularly Convolutional Neural Networks (CNNs) and transfer learning—for the accurate identification of plant diseases across multiple crop species using leaf images.

Additional specific objectives include:

- To implement and experiment with foundational neural network models for image classification.
- To fine-tune pre-trained architectures such as EfficientNet-B0 Tan and Le (2019) and MobileNetV2 Sandler et al. (2018) for use in agricultural disease detection tasks.
- To curate, clean, and balance a multi-class, multi-crop dataset suitable for disease classification.
- To evaluate the trained models using metrics such as accuracy, confusion matrices, and per-class performance.
- To validate the system under real-world image conditions and document its limitations and robustness.
- To demonstrate the feasibility of lightweight, mobile-friendly AI solutions for use by farmers in real-time field scenarios.

2 SYSTEM REQUIREMENTS

The system was developed and tested in a cloud-based Python environment using Google Colab. Below are the software and hardware requirements used during the training and testing phases.

Software Requirements

- Programming Language: Python 3.8+
- IDE/Platform: Google Colab
- Libraries and Frameworks:
  - TensorFlow

  - Keras
  - NumPy
  - Matplotlib
  - scikit-learn
  - OpenCV

- Dataset Source: Kaggle (PlantVillage, Mendeley datasets)

Hardware Requirements

- CPU: Intel i5/i7 or equivalent (cloud-hosted)
- GPU: Tesla K80/TPU v2 (via Google Colab)
- RAM: 12GB+ (Colab runtime)
- Storage: Approx. 2GB for dataset and models

Google Colab was primarily used to access free GPU/TPU resources and manage model training pipelines efficiently.

3 SYSTEM ARCHITECTURE

The architecture of the plant disease detection system follows a modular pipeline. Each module is responsible for one specific function, ranging from data handling to model evaluation. The primary components of the system are:

1. **Dataset Loading:** Images are loaded from a curated and cleaned version of the plant leaves dataset, containing labeled leaf images from various crops and disease classes.

2. **Preprocessing:** Images are resized, normalized, and augmented using standard techniques such as flipping, rotation, and brightness adjustment.

3. **Model Training:**

   - Two types of transfer learning models are trained:

     (a) A model for each of the five plants.

     (b) A combined model to detect disease for all 5 plants.

   - Every model is compiled using the Adam optimizer and categorical cross-entropy loss.

4. **Evaluation:** The models are evaluated using accuracy, validation loss, confusion matrices, and classification reports.

5. **Prediction:** The trained model is used to predict new leaf images and visualize results via class labels.

Layer (type)	Output Shape	Param #
input_layer_1 (InputLayer)	(None, 224, 224, 3)	0
mobilenetv2_1.00_224 (Functional)	(None, 7, 7, 1280)	2,257,984
global_average_pooling2d (GlobalAveragePooling2D)	(None, 1280)	0
dense (Dense)	(None, 32)	40,992
batch_normalization (BatchNormalization)	(None, 32)	128
dropout (Dropout)	(None, 32)	0
dense_1 (Dense)	(None, 6)	198

Total params: 2,299,302 (8.77 MB)

Trainable params: 41,254 (161.15 KB)

Non-trainable params: 2,258,048 (8.61 MB)

Figure 2. System architecture of MobileNetV2-based model for individual crops

Layer (type)	Output Shape	Param #
input_layer_1 (InputLayer)	(None, 224, 224, 3)	0
mobilenetv2_1.00_224 (Functional)	(None, 7, 7, 1280)	2,257,984
global_average_pooling2d (GlobalAveragePooling2D)	(None, 1280)	0
dense (Dense)	(None, 128)	163,968
batch_normalization (BatchNormalization)	(None, 128)	512
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 28)	3,612

Total params: 2,426,076 (9.25 MB)

Trainable params: 167,836 (655.61 KB)

Non-trainable params: 2,258,240 (8.61 MB)

Figure 3. System architecture of MobileNetV2-based model for combined crops

4 DATASET DESCRIPTION

The dataset used in this study is a custom-curated compilation based on publicly available plant disease image repositories, primarily **PlantVillage** and **Mendeley Data**. It includes leaf images for five major crops — **corn, grape, mango, peanut, and tomato** — with both healthy and disease-affected categories represented.

Each image is an **RGB JPG file** organized in a folder structure where the root folder represents the crop name, and subfolders represent disease classes.

4.1 Dataset Structure and Class Distribution

The final dataset comprises **28 distinct classes**, with **1200 images per class**, making it a balanced dataset. This balance was achieved using:

- **Oversampling:** For underrepresented classes, synthetic images were generated by applying rotations of 90°, 180°, and 270°.
- **Undersampling:** For overrepresented classes, random images were deleted to maintain class uniformity.

Class Categories:

- **Corn:** Blight, Common Rust, Gray Leaf Spot, Healthy
- **Grape:** Black Rot, Esca, Leaf Blight, Healthy
- **Mango:** Gall Midge, Healthy, Powdery Mildew, Sooty Mould
- **Peanut:** Early Leaf Spot, Early Rust, Healthy Leaf, Late Leaf Spot, Nutrition Deficiency, Rust
- **Tomato:** Bacterial Spot, Early Blight, Late Blight, Leaf Mold, Septoria Leaf Spot, Spider Mites (Twospotted), Target Spot, Yellow Leaf Curl Virus, Mosaic Virus, Healthy

Total Images: 33,600
Classes: 28
Images per Class: 1200

4.2 Train-Validation-Test Split

The dataset was divided using stratified sampling to preserve class balance:

- Training: 80%
- Validation: 10%
- Testing: 10%

4.3 Data Augmentation

To increase model robustness and prevent overfitting, the following augmentation techniques were applied to the training set using Keras's `ImageDataGenerator`:

```
ImageDataGenerator(
    rescale=1./255,
    rotation_range=40,
    width_shift_range=0.3,
    height_shift_range=0.3,
    brightness_range=[0.5, 1.5],
    shear_range=0.2,
    zoom_range=0.3,
    channel_shift_range=0.2,
    fill_mode='reflect',
    horizontal_flip=True
)
```

These augmentations simulate lighting variations, scale shifts, geometric distortions, and leaf orientation changes.

4.4 Source and Access

The original data was sourced from:

- PlantVillage Dataset Mohanty et al. (2016)
- Mendeley Datasets Ahmad (2025) Solapure et al. (2024) Manvikar and Reddy (2023) Rahman et al. (2024)

The final curated version of this dataset can be accessed at the following link:

Dataset Link: https://www.kaggle.com/datasets/ankurpaul52/balanced-multi-crop-plant-disease-dataset

Figure 4. Sample images from the training dataset

5 METHODOLOGY

5.1 Data Preprocessing

Before training the model, several preprocessing steps were applied to prepare the raw image data and make it suitable for input into deep learning models:

- **Image Resizing:** All images were resized to a fixed resolution of `224 × 224` pixels, which is the input size required by CNN architectures such as EfficientNet-B0 and MobileNetV2.
- **RGB Conversion:** To maintain consistency, all images were ensured to be in `RGB format`.
- **Pixel Value Rescaling:** Pixel intensities, originally in the range `[0, 255]`, were normalized to the range `[0, 1]` using the formula:

$$\text{normalized pixel} = \frac{\text{original pixel}}{255}$$

These preprocessing operations were applied consistently to the training, validation, and test datasets.

5.2 Model 1: Individual Model for each of the five crops

To begin experimentation, a separate classification model was trained for each of the five crops: corn, grape, mango, peanut, and tomato. This allowed us to evaluate how well transfer learning works in simpler scenarios involving fewer classes per model.

For each crop:

- The dataset was filtered to include only the images relevant to that specific crop.

- The number of classes ranged from 4 to 10, depending on the crop.
- The architecture used was **MobileNetV2**, initialized with pre-trained ImageNet weights and fine-tuned on the crop-specific data.
- A csutom classification head, as described in Figure 2, was used.
- The loss function used was categorical cross-entropy.

This approach helped verify the feasibility of accurate disease detection per crop and acted as a baseline to compare with the combined model.

5.3 Model 2: Combined Model for all 5 crops (28 classes)

After successfully training crop-specific models, a unified multi-class model was developed to classify all 28 categories across the five crops.

- The complete dataset, containing all five crops and 28 classes, was used.
- **MobileNetV2** was again used as the base architecture with ImageNet weights.
- A custom classification head was added, as described in Figure 3:
- The loss function used was categorical cross-entropy.
- Data augmentation was extensively used to improve generalization.

This combined model posed a more challenging task due to visual similarities across diseases and crops.

5.4 Training Details

Transfer learning was applied using the pre-trained MobileNetV2 architecture to classify diseases across balanced crop datasets. Training consisted of two phases:

1. **Feature Extraction (Frozen Phase)**: In the initial phase, the convolutional base of MobileNetV2 was frozen, and only the newly added classification head was trained. This allowed the model to learn task-specific features without altering the pre-trained weights (Chollet and the Keras Team (2023)). Training during this phase was conducted for **10 epochs for individual crop models** and **15 epochs for the combined model**.

2. **Fine-Tuning Phase**: After the classification head was trained, the top 20 layers of MobileNetV2 were unfrozen and jointly trained along with the head layers. This allowed for fine adjustments to the feature representations. Fine-tuning was also performed for **10 epochs (individual models)** and **15 epochs (combined model)**.

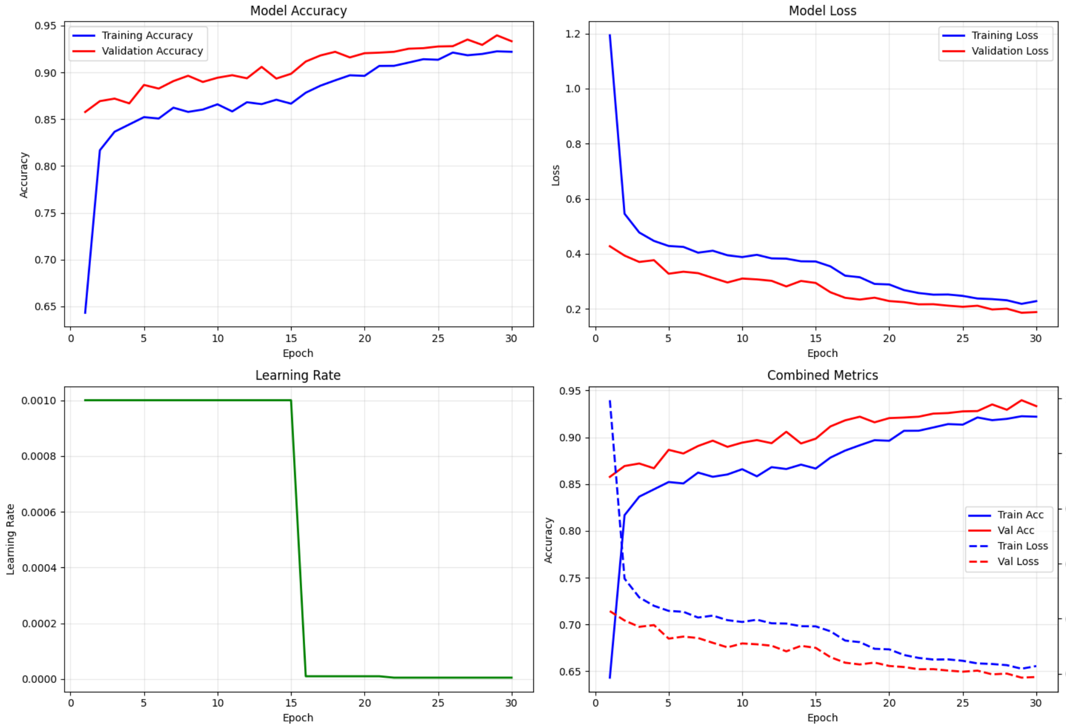


Figure 5. Training and validation accuracy, and loss curves for the combined model.

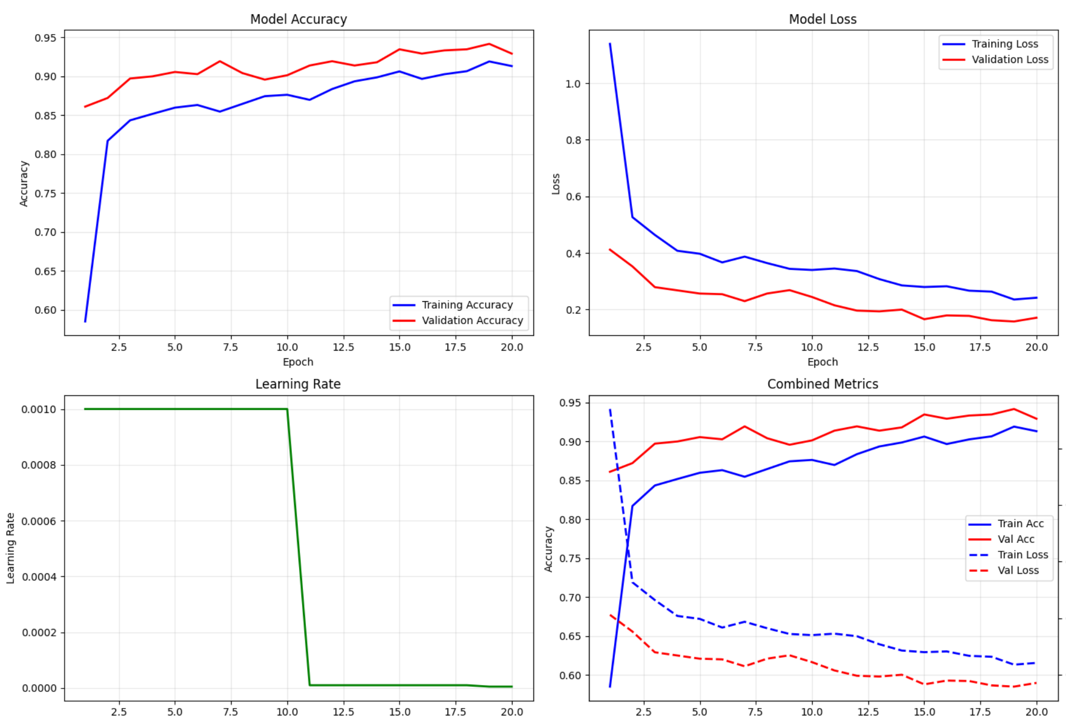


Figure 6. Training and validation accuracy, and loss curves for the Peanut model

6 RESULTS AND EVALUATION

6.1 Training Accuracy and Loss

Table 1. Training Accuracy and Loss for individual and combined models

Model	Training Loss	Training Accuracy (%)
Corn	0.2656	89.84
Peanut	0.2217	92.04
Grape	0.0718	97.06
Mango	0.1197	95.44
Tomato	0.4523	85.50
Combined	**0.2268**	**91.82**

6.2 Validation Accuracy

Table 2. Validation accuracy for individual and combined models

Model	Val Accuracy (%)
Corn	92.71
Peanut	95.70
Grape	98.33
Mango	98.54
Tomato	88.00
Combined	**95.15**

6.3 Confusion Matrix

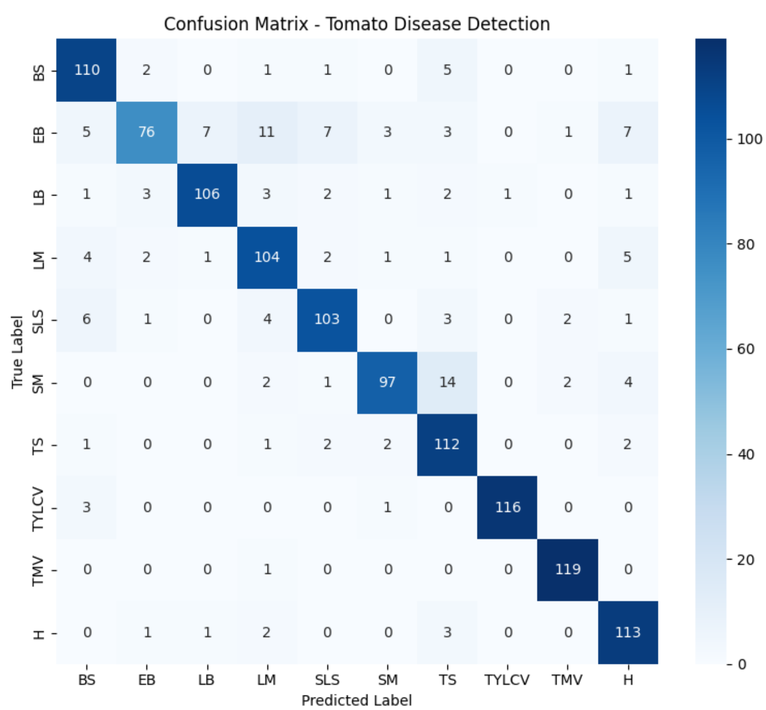
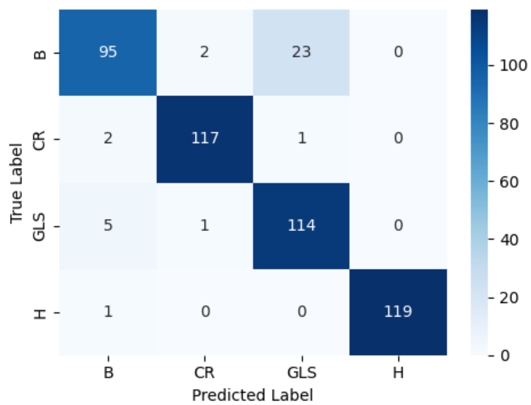
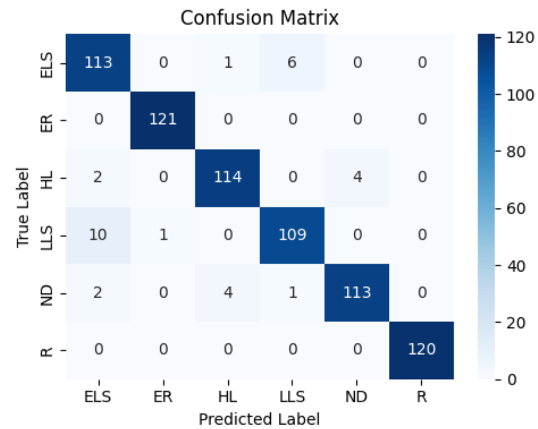


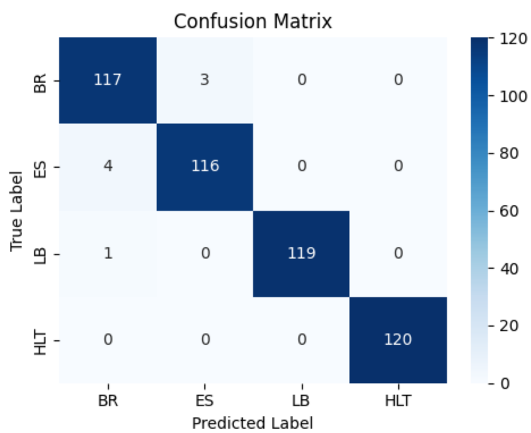
Figure 7. Confusion matrix for the tomato model



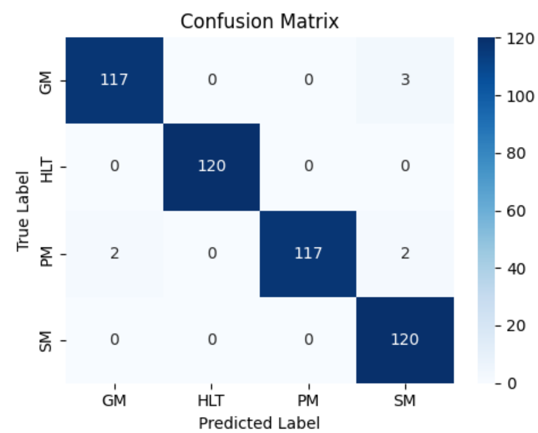
(a) Corn model



(b) Peanut model



(c) Grape model



(d) Mango model

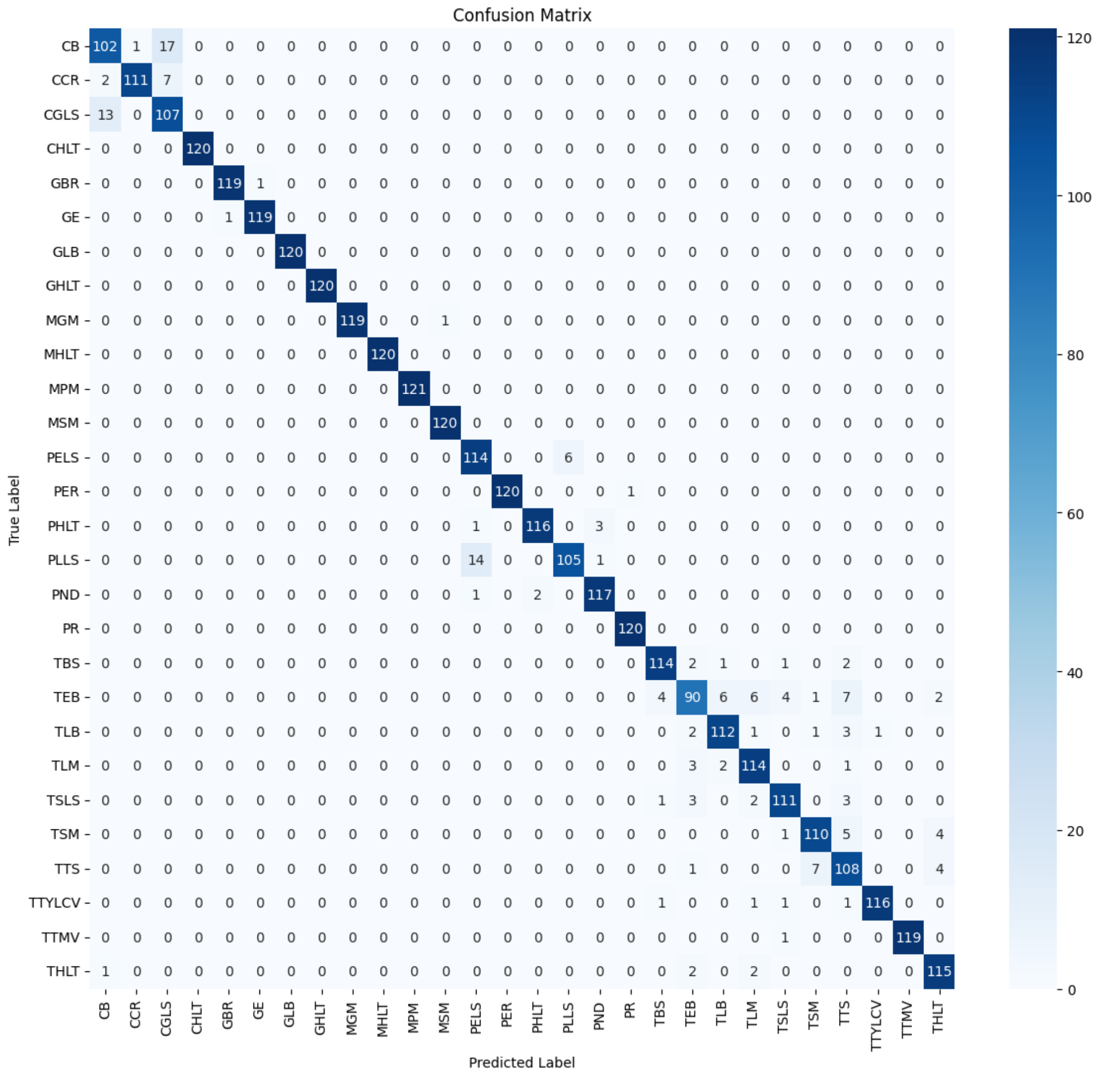


Figure 9. Confusion matrix for the combined model

6.4 Graphical Analysis

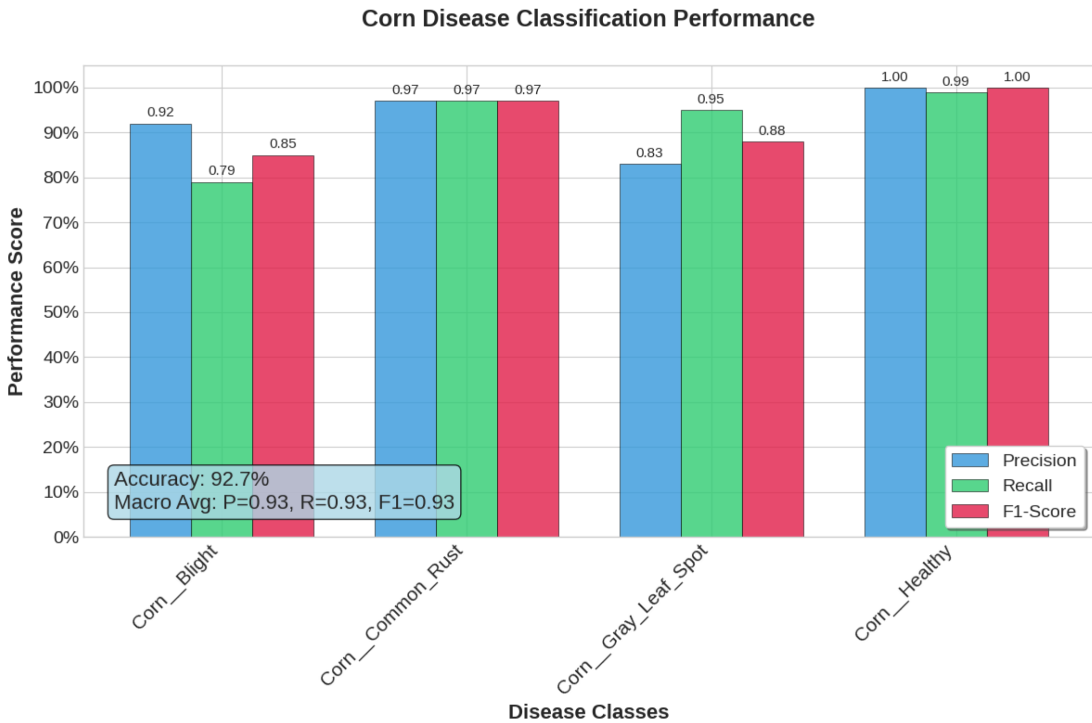


Figure 10. Precision, Recall, and F1-score comparison for Corn model

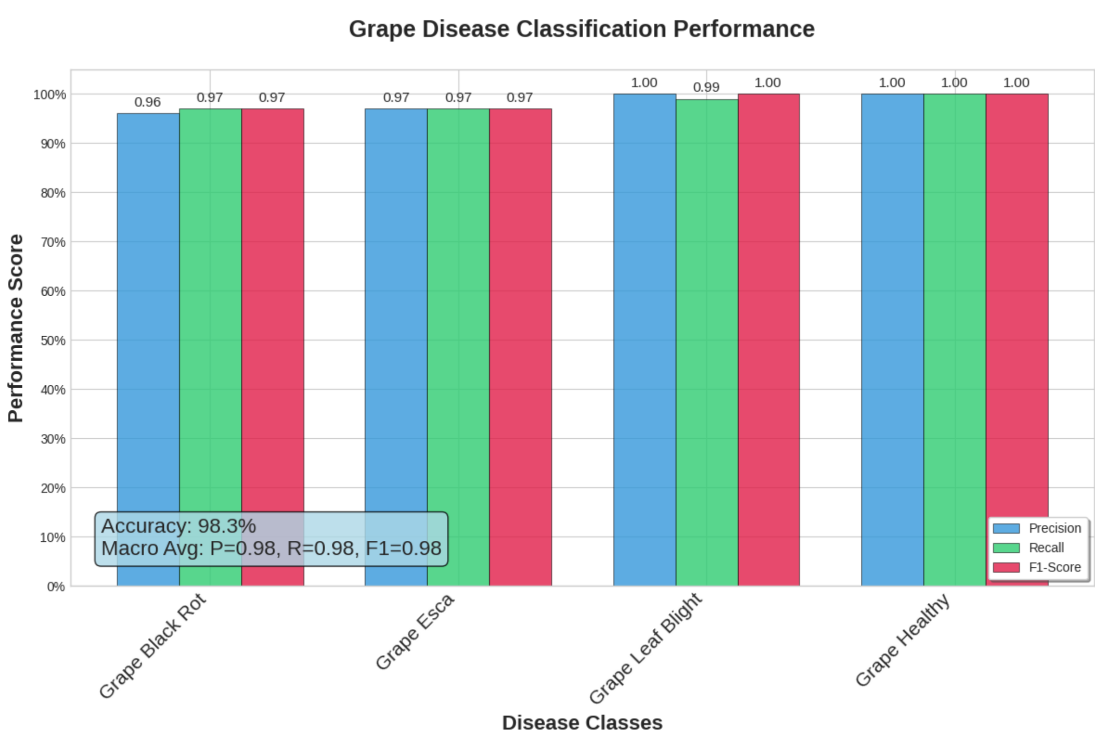


Figure 11. Precision, Recall, and F1-score comparison for Grape model

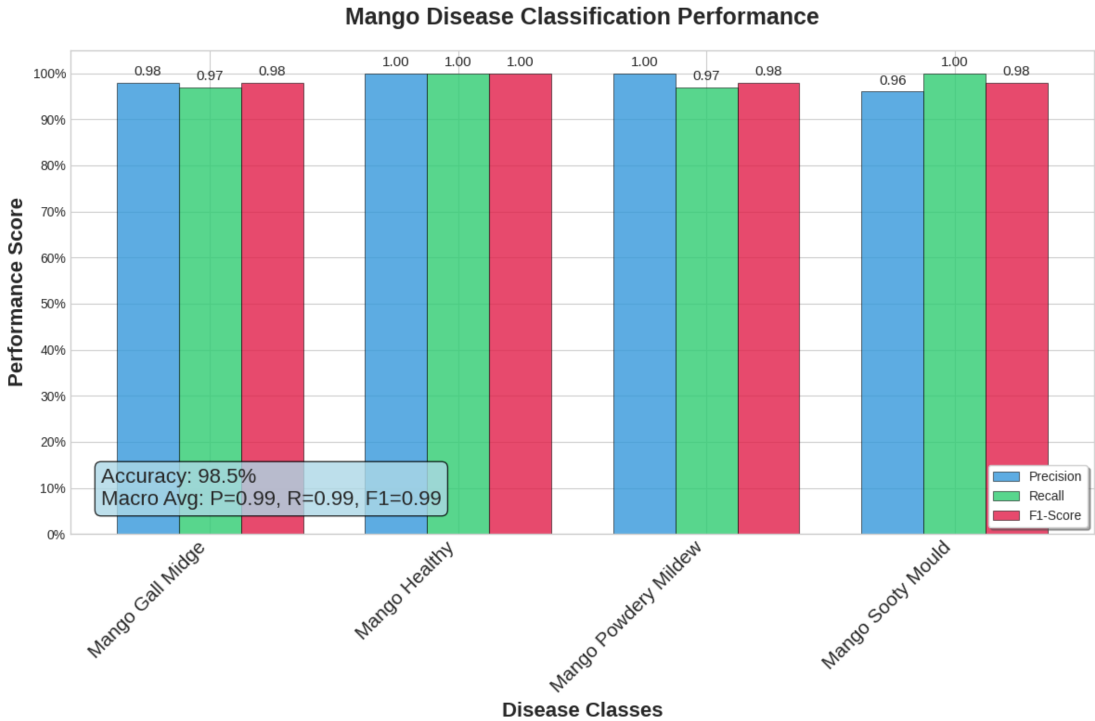


Figure 12. Precision, Recall, and F1-score comparison for Mango model

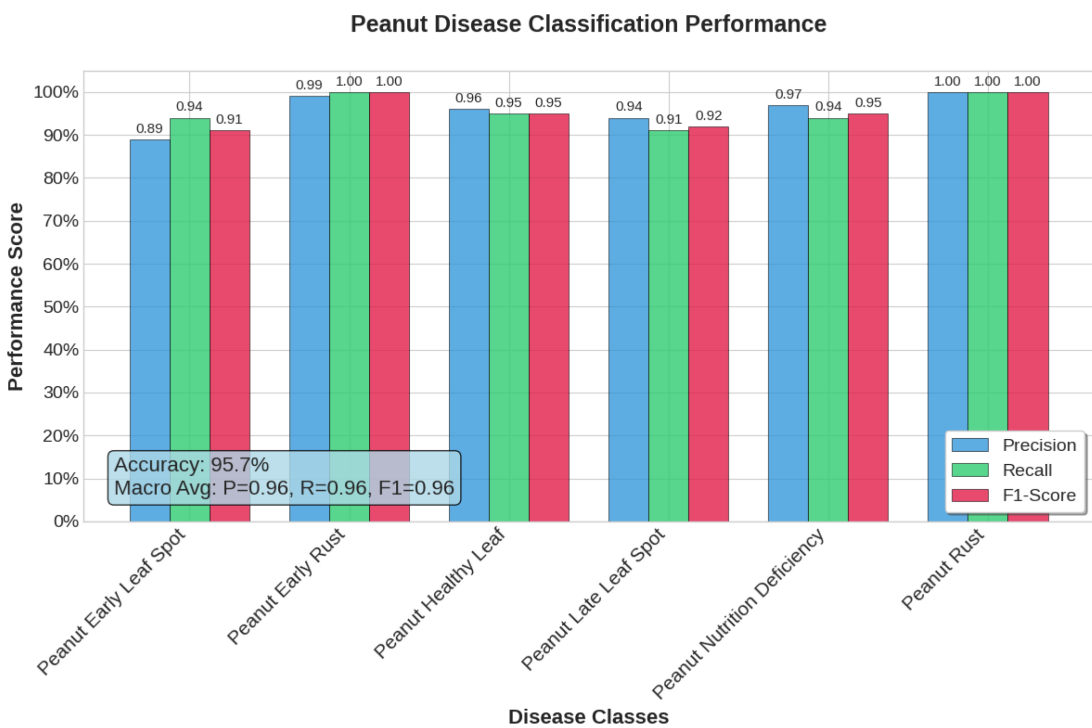


Figure 13. Precision, Recall, and F1-score comparison for Peanut model

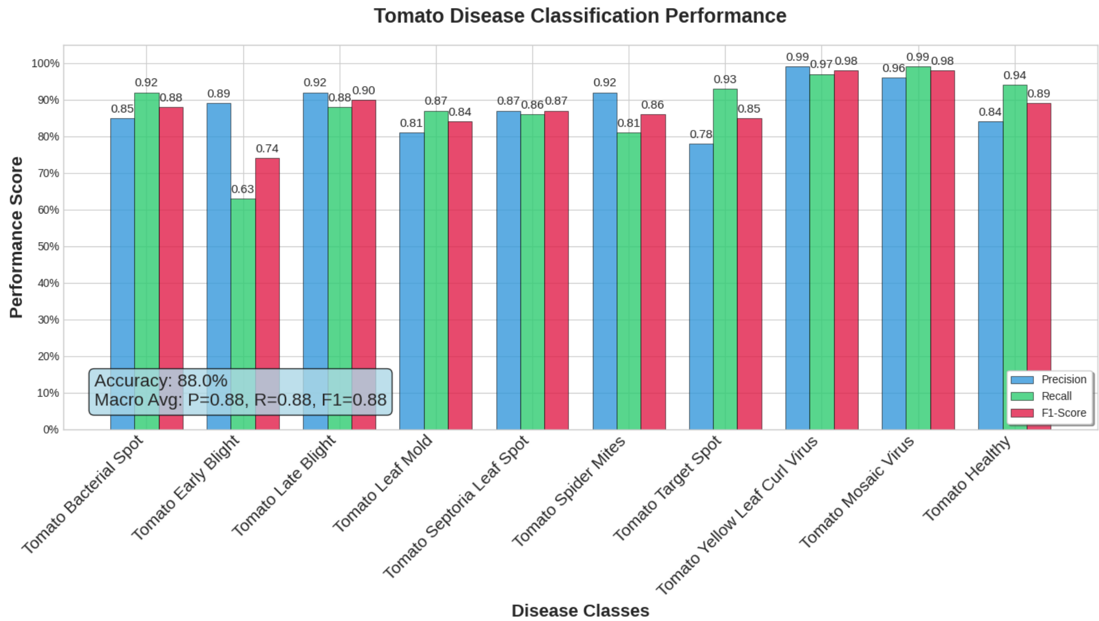


Figure 14. Precision, Recall, and F1-score comparison for Tomato model

Table 3. Classification report for the combined (28 classes)

Class	**Precision**	**Recall**	**F1-Score**	**Support**
Corn__Blight	0.91	0.88	0.89	120
Corn__Common_Rust	0.99	0.96	0.97	120
Corn__Gray_Leaf_Spot	0.87	0.92	0.89	120
Corn__Healthy	0.99	1.00	1.00	120
Grape__Black_rot	0.99	0.96	0.97	120
Grape__Esca	0.95	1.00	0.98	120
Grape__Leaf_blight	0.99	0.99	0.99	120
Grape__healthy	1.00	1.00	1.00	120
Mango__Gall_Midge	0.99	0.98	0.99	120
Mango__Healthy	1.00	1.00	1.00	120
Mango__Powdery_Mildew	1.00	1.00	1.00	121
Mango__Sooty_Mould	0.98	1.00	0.99	120
Peanut__early_leaf_spot	0.93	0.96	0.94	120
Peanut__early_rust	1.00	1.00	1.00	121
Peanut__healthy_leaf	0.95	0.99	0.97	120
Peanut__late_leaf_spot	0.95	0.93	0.94	120
Peanut__nutrition_deficiency	0.99	0.94	0.97	120
Peanut__rust	1.00	1.00	1.00	120
Tomato__Bacterial_spot	0.97	0.93	0.94	120
Tomato__Early_blight	0.85	0.78	0.82	120
Tomato__Late_blight	0.97	0.96	0.96	120
Tomato__Leaf_Mold	0.87	0.93	0.90	120
Tomato__Septoria_leaf_spot	0.90	0.95	0.93	120
Tomato__Spider_mites_Twospotted_spider_mite	0.95	0.87	0.91	120
Tomato__Target_Spot	0.83	0.94	0.88	120
Tomato__Tomato_Yellow_Leaf_Curl_Virus	0.99	0.94	0.97	120
Tomato__Tomato_mosaic_virus	0.98	0.98	0.98	120
Tomato__healthy	0.93	0.94	0.94	120
Accuracy		0.95		3362
Macro Avg	0.96	0.95	0.95	3362
Weighted Avg	0.96	0.95	0.95	3362

7 TESTING AND SAMPLE PREDICTIONS

Figure 15 shows 3×3 grids of predicted vs. true labels on validation images.

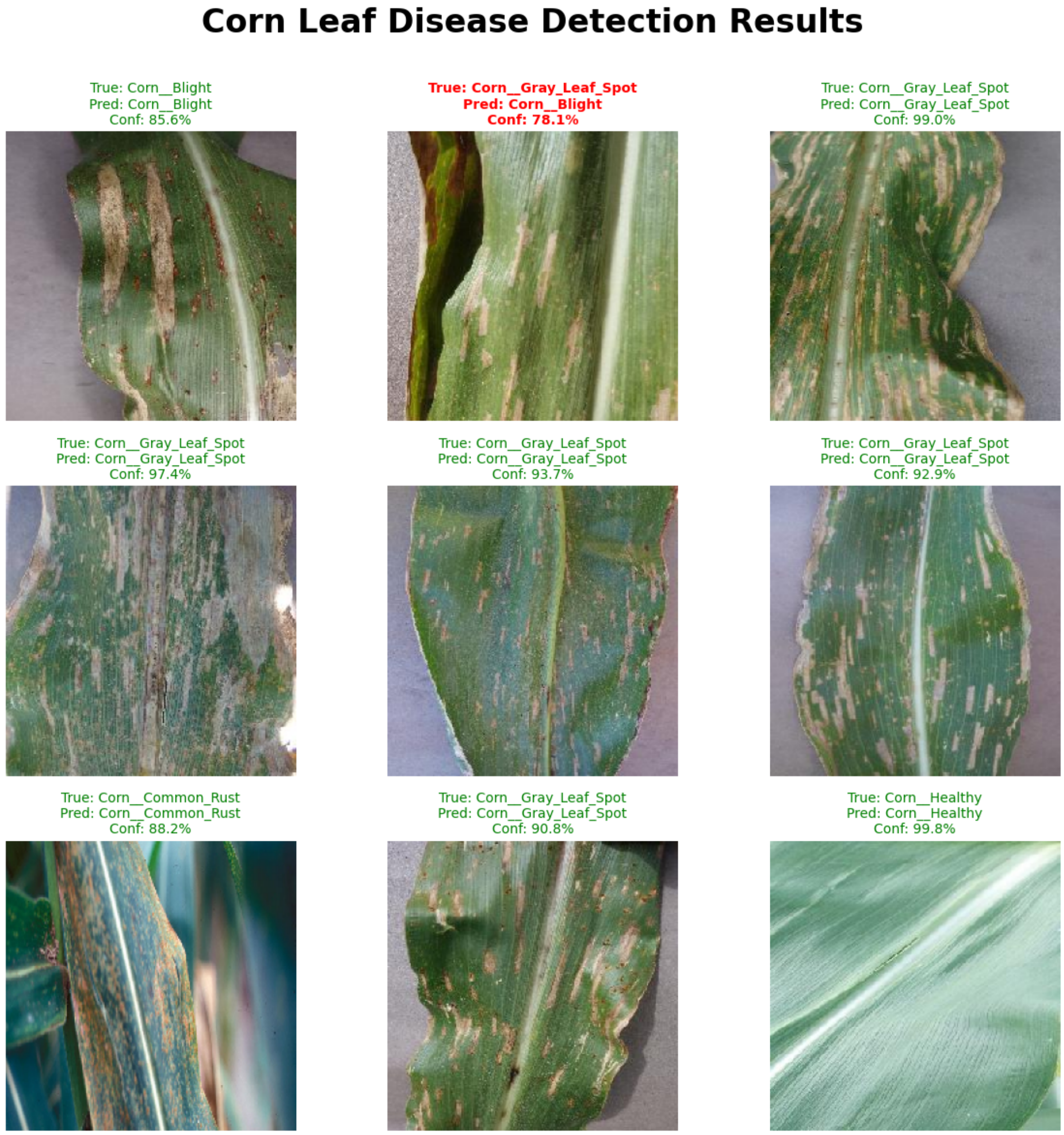


Figure 15. Predictions on validation images for corn model

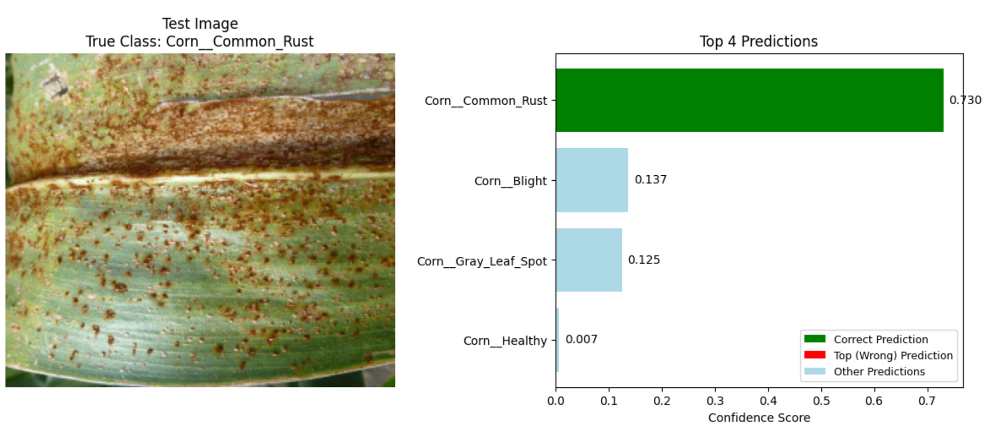


Figure 16. Real-world test images for corn disease detection

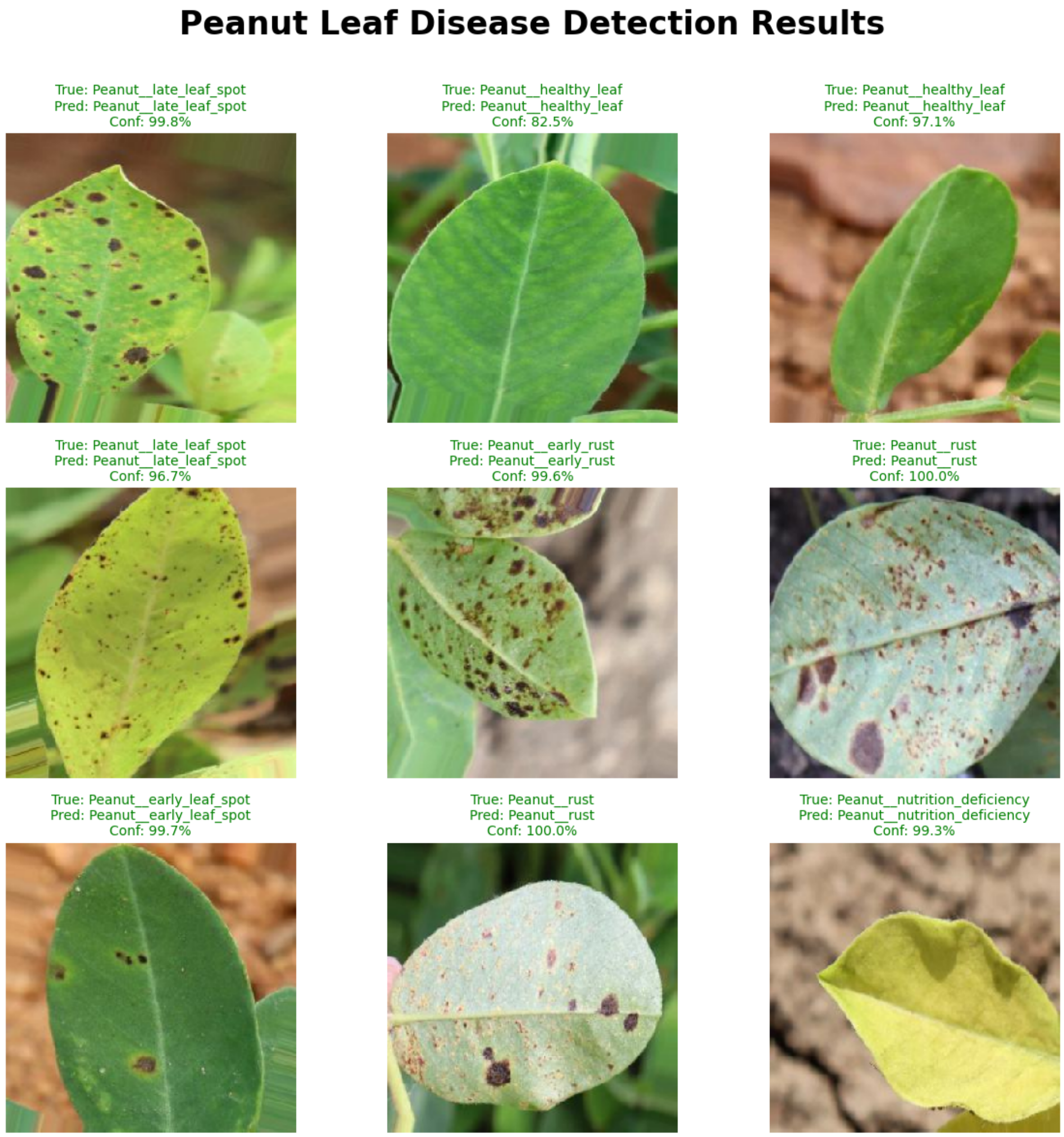


Figure 17. Predictions on validation images for peanut model

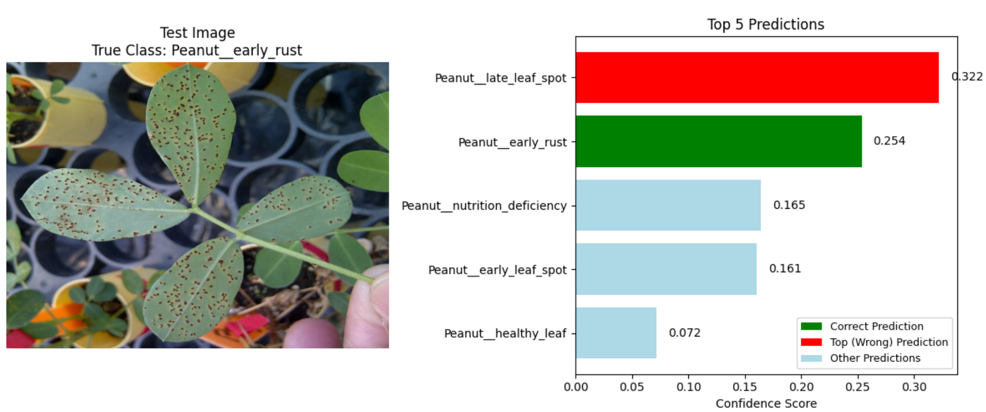


Figure 18. Real-world test images for peanut disease detection

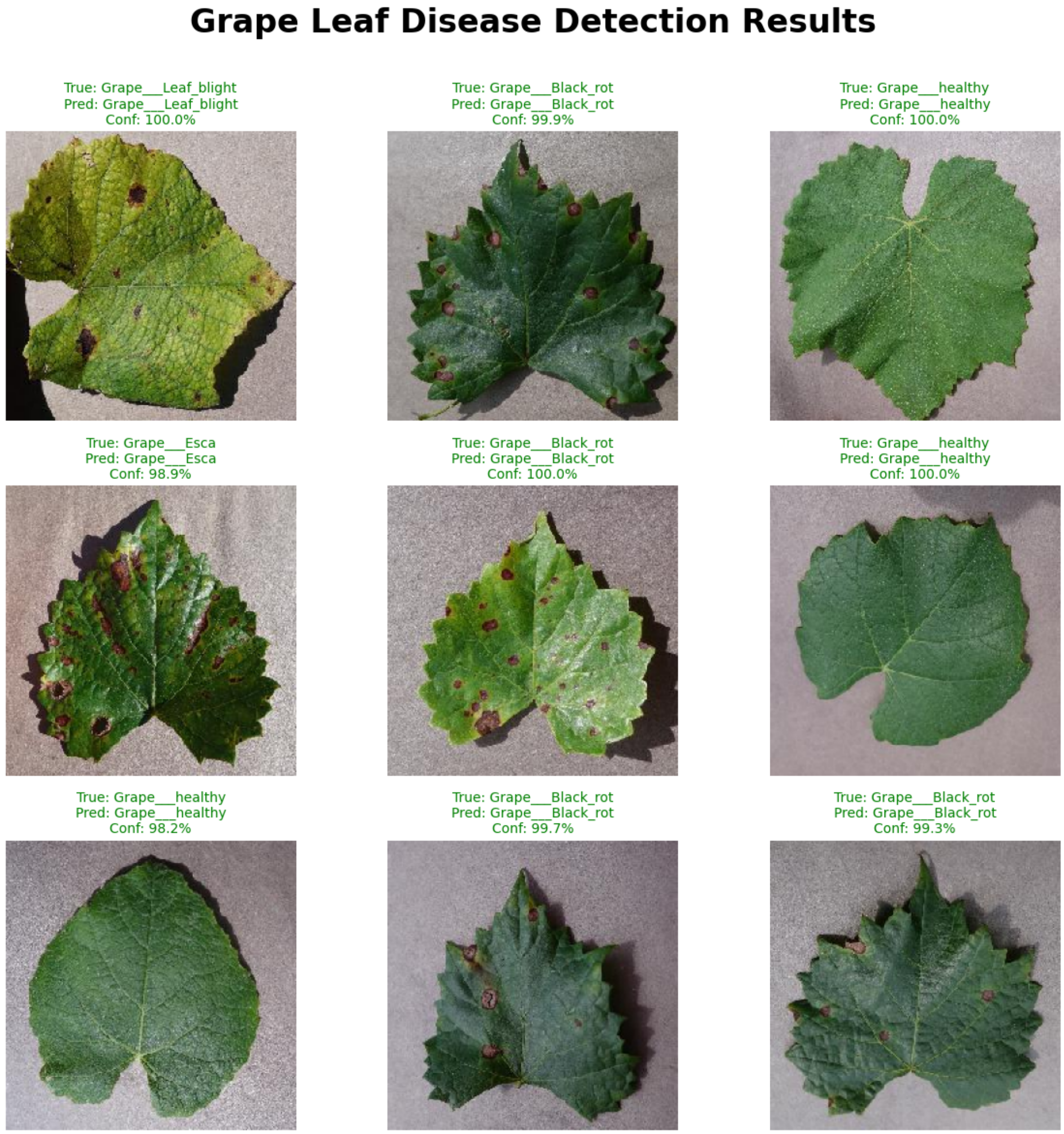


Figure 19. Predictions on validation images for grape model

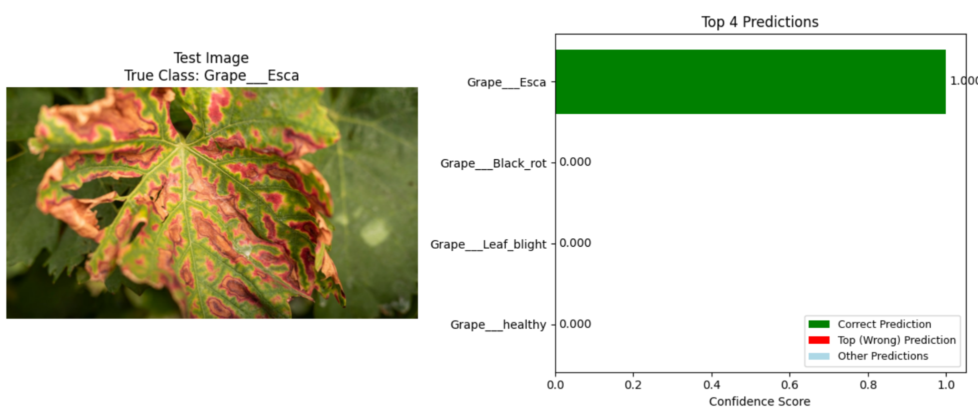


Figure 20. Real-world test images for grape disease detection

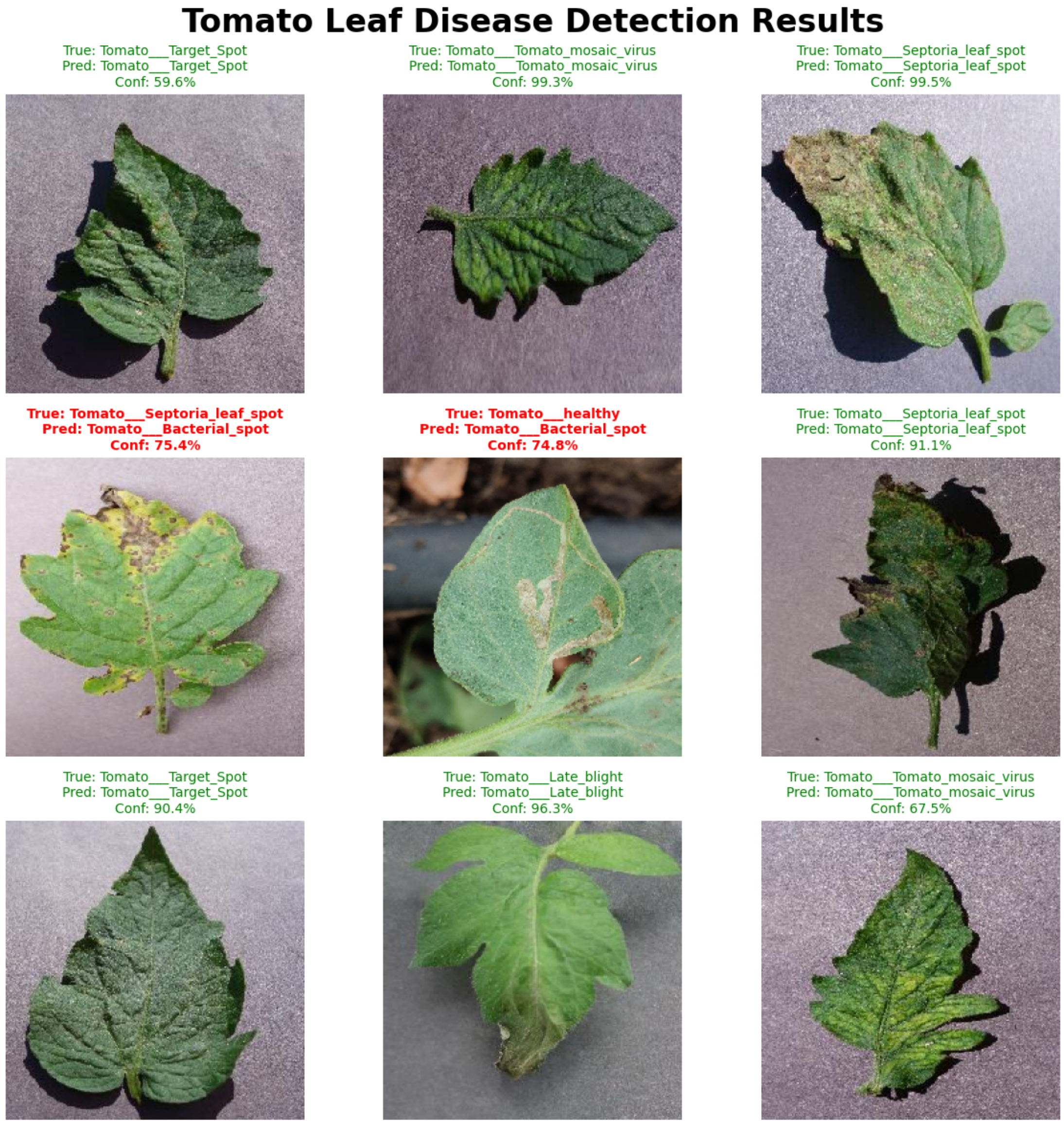


Figure 21. Predictions on validation images for tomato model

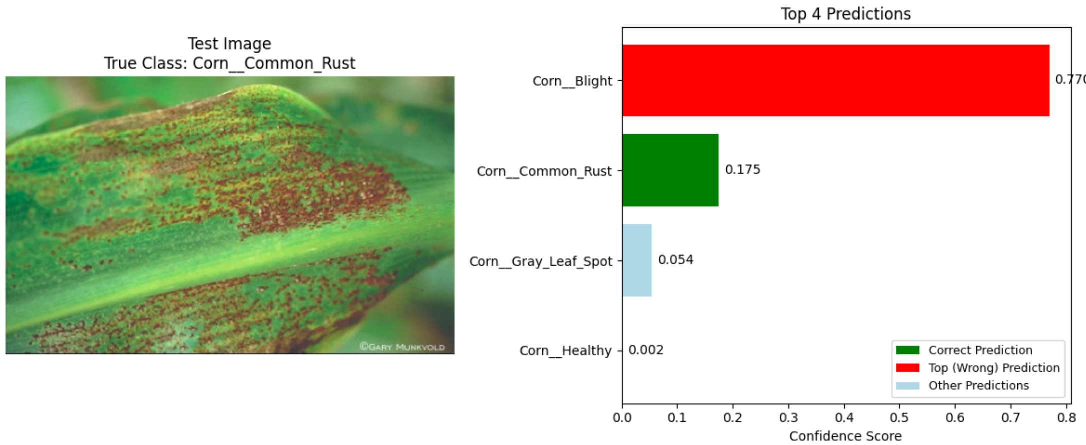


Figure 22. Real-world test images for corn disease detection

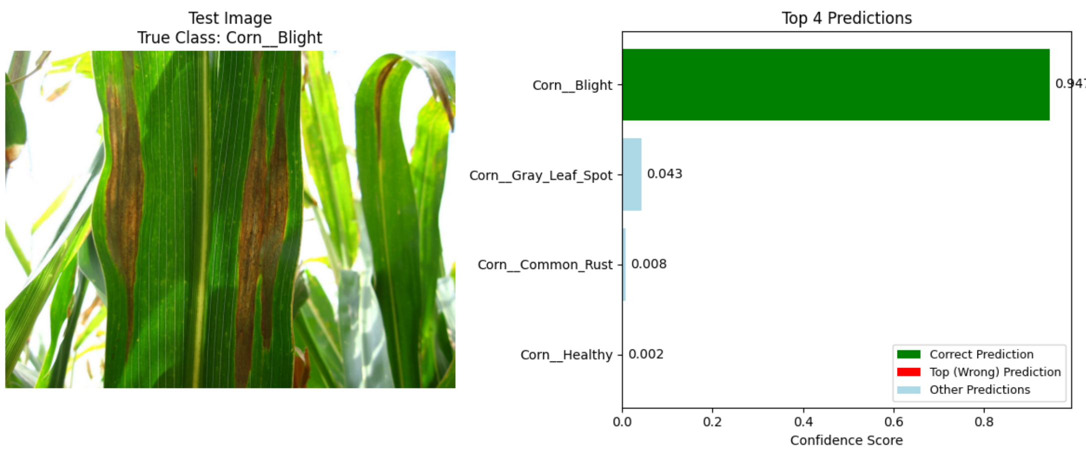


Figure 23. Real-world test images for corn disease detection

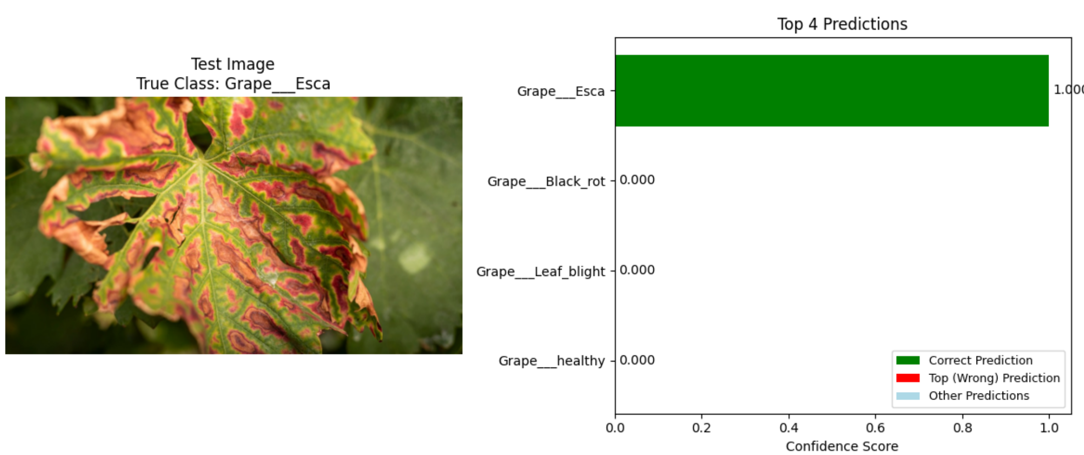


Figure 24. Real-world test images for grape disease detection

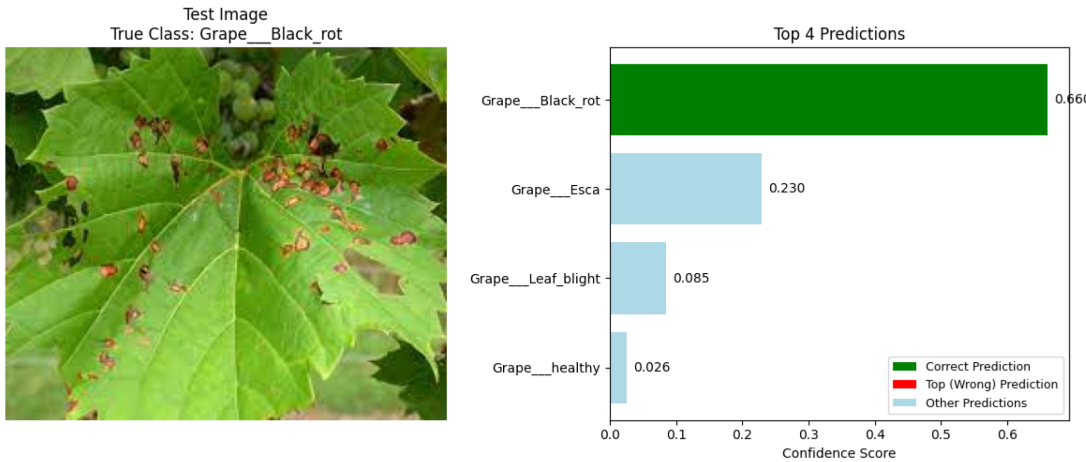


Figure 25. Real-world test images for grape disease detection

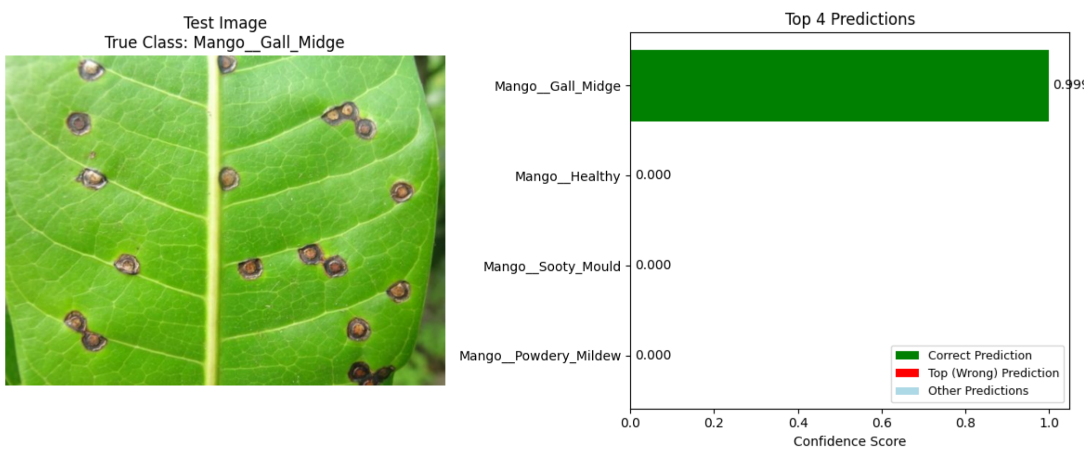


Figure 26. Real-world test images for mango disease detection

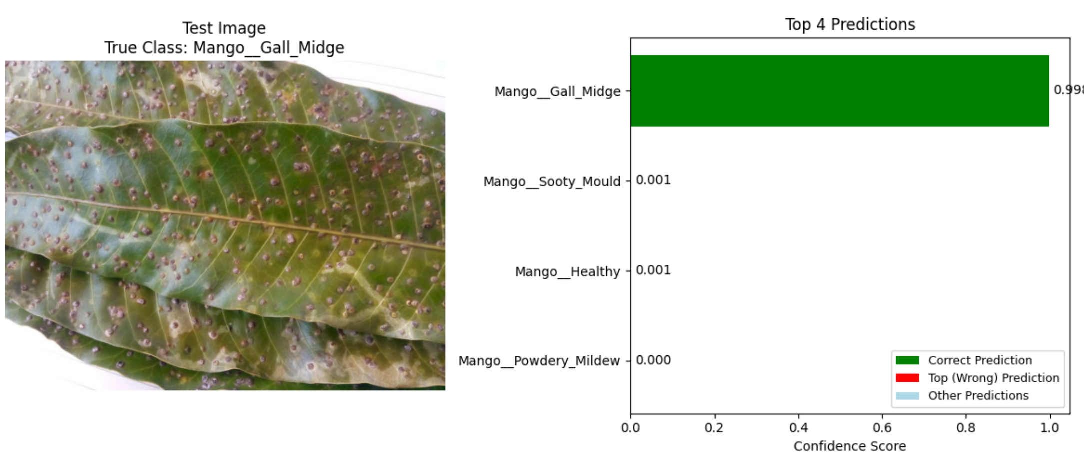


Figure 27. Real-world test images for mango disease detection

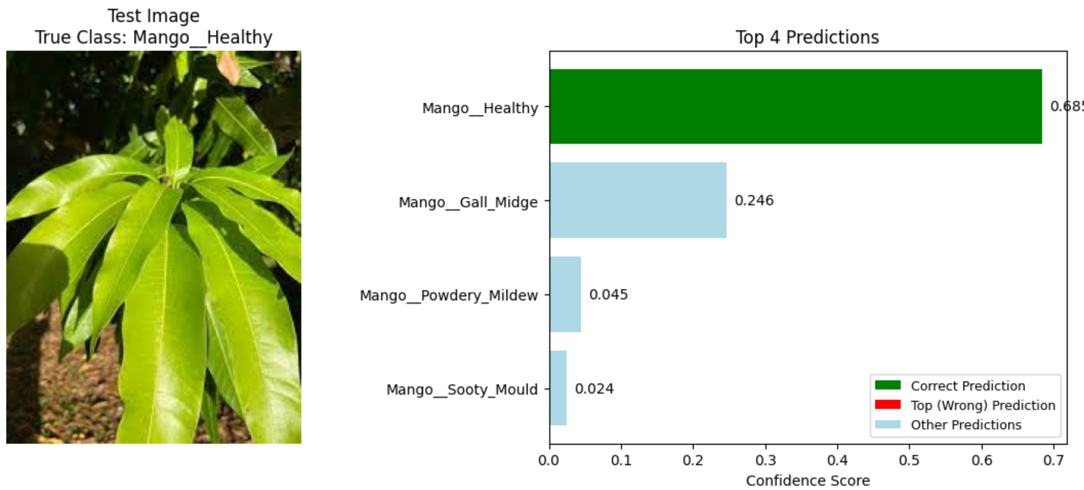


Figure 28. Real-world test images for mango disease detection

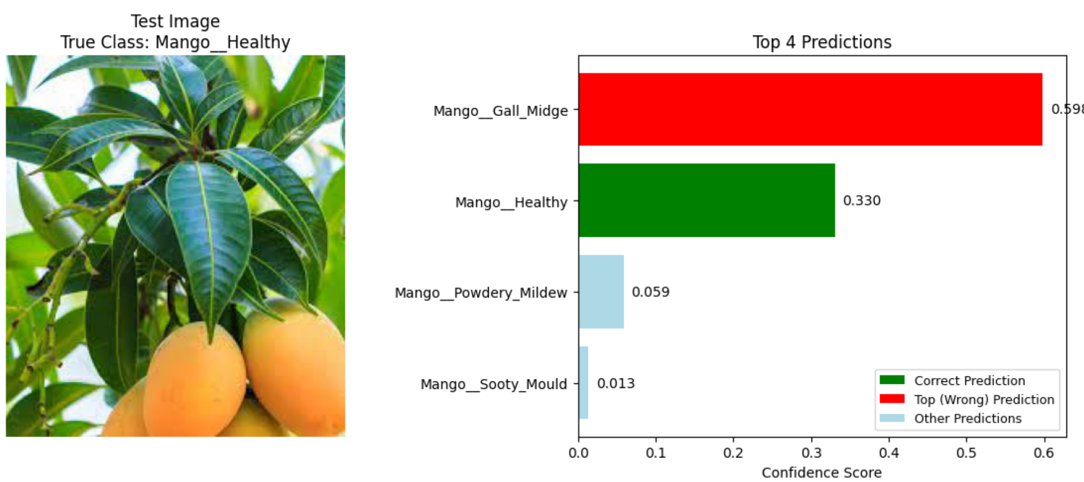


Figure 29. Real-world test images for mango disease detection

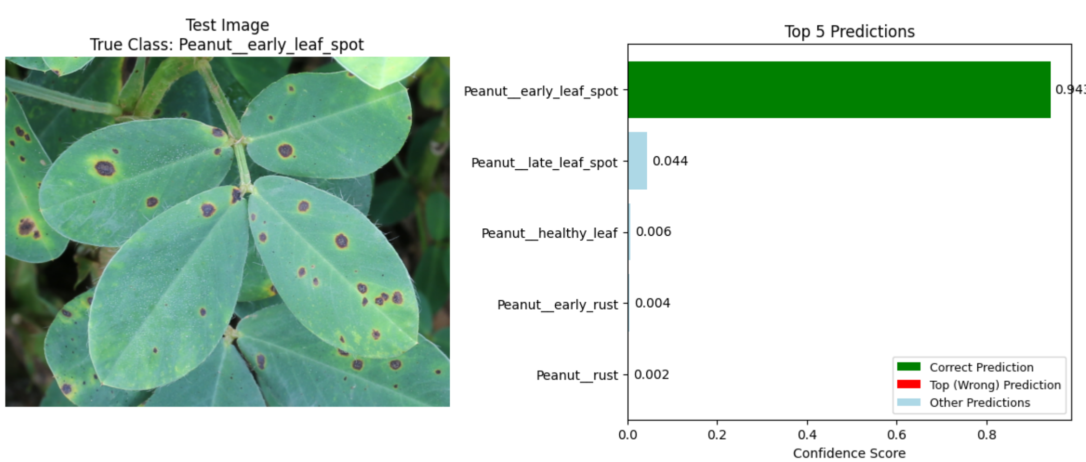


Figure 30. Real-world test images for peanut disease detection

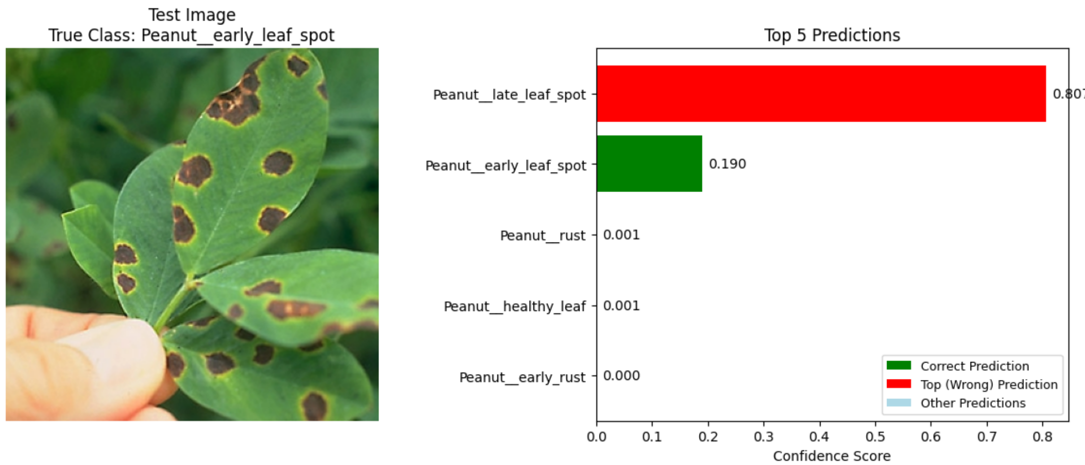


Figure 31. Real-world test images for peanut disease detection

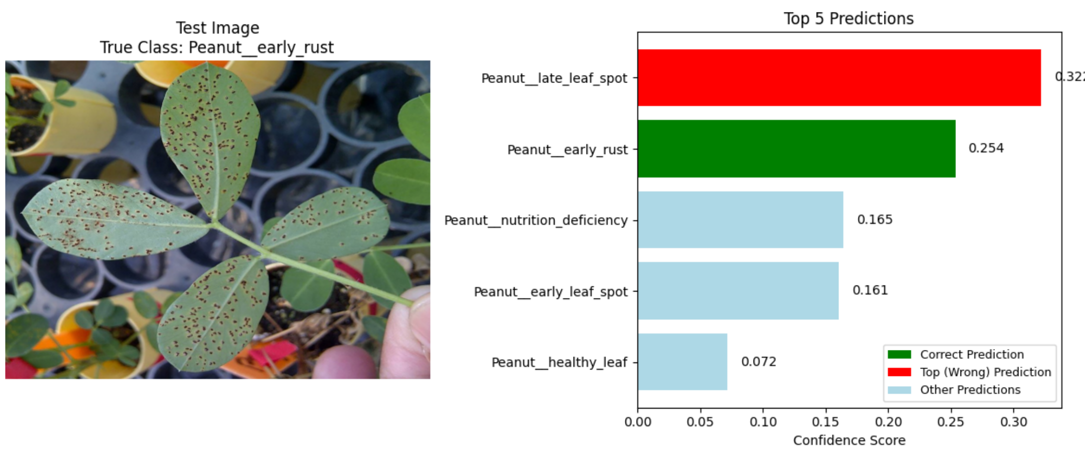


Figure 32. Real-world test images for peanut disease detection

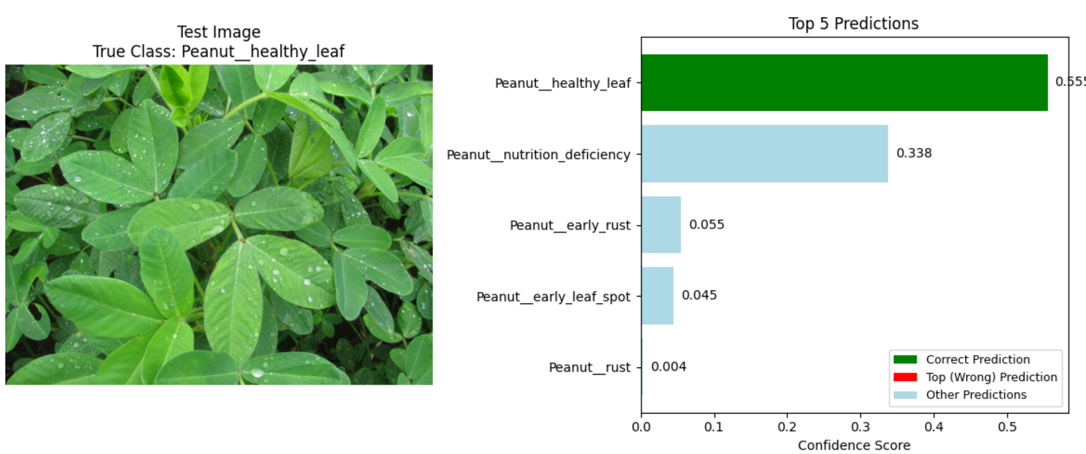


Figure 33. Real-world test images for peanut disease detection

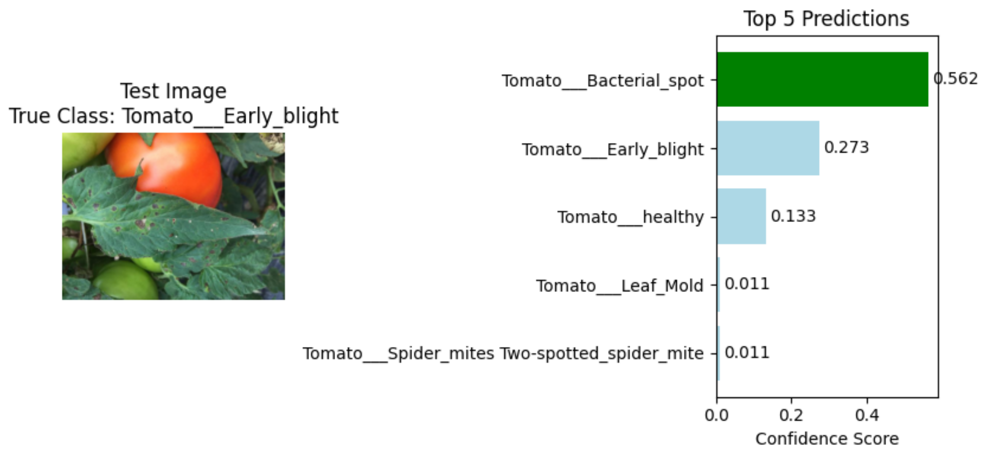


Figure 34. Real-world test images for tomato disease detection

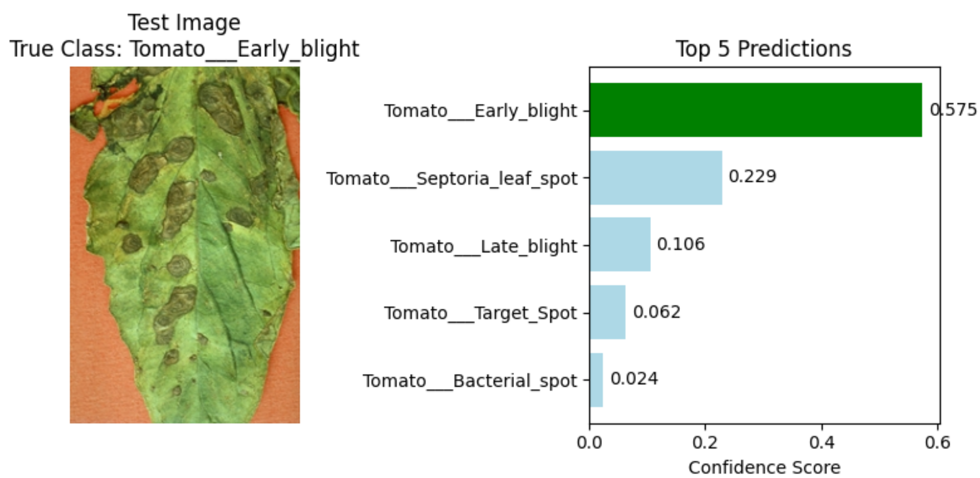


Figure 35. Real-world test images for tomato disease detection

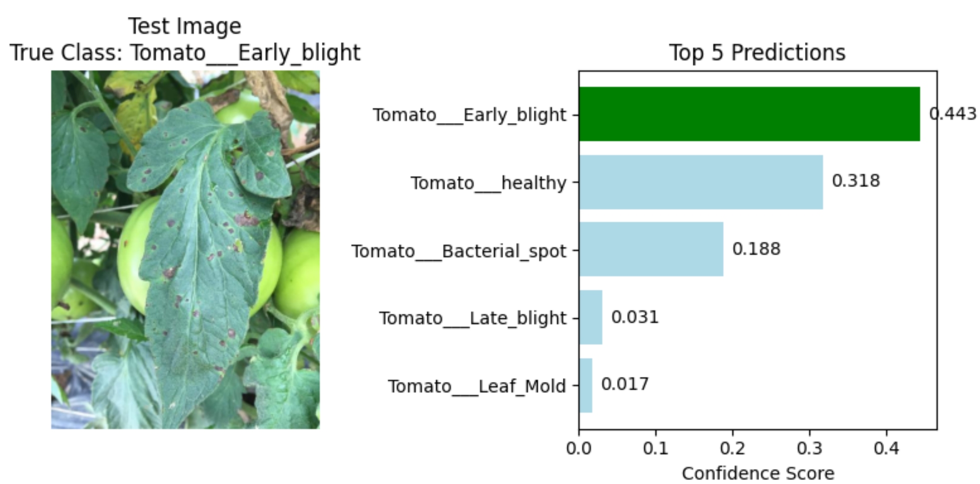


Figure 36. Real-world test images for tomato disease detection

8 APPLICATIONS

The proposed plant disease detection system has significant practical and research applications:

1. **Mobile-Based Field Diagnosis:** The lightweight MobileNetV2 architecture supports real-time inference on smartphones and edge devices, making it ideal for mobile applications in field conditions. Farmers can capture leaf images and get disease predictions instantly.

2. **Farmer Decision Support:** By providing early and accurate disease identification, the system empowers farmers to take timely actions such as pesticide use, nutrient correction, or crop isolation, reducing yield loss and promoting precision agriculture.

3. **Agricultural Extension Services:** The model can assist agricultural officers and field workers as a preliminary disease detection tool, especially in remote or rural areas where access to experts may be limited.

4. **Integration into Smart Farming Systems:** The system can be incorporated into Internet of Things (IoT)-based platforms to enable automated monitoring, alert generation, and field-level disease tracking.

5. **Educational Tool:** This system can be used in academic settings to teach students and researchers about real-world applications of deep learning in agriculture, including dataset curation and transfer learning.

9 LIMITATIONS

Despite achieving high validation accuracy, the system has the following limitations:

1. **Single-Label Classification:** The system is currently restricted to single-label classification and cannot identify multiple co-existing diseases on a single leaf, which is a common real-world scenario.

2. **Combined Model Confusion:** The unified 28-class model occasionally misclassifies diseases that appear visually similar across crops, showing reduced performance compared to individual crop models.

3. **Limited Dataset Diversity:** Although balanced and augmented, the dataset is derived from controlled sources and lacks variations seen in field environments, such as different leaf ages, damage levels, and seasonal changes.

4. **Lack of Contextual Features:** The model only relies on visual features from leaf images and does not incorporate agronomic or environmental data (e.g., humidity, soil health), which could improve its decision-making ability.

10 FUTURE SCOPE

While the current model exhibits strong performance on single-leaf, single-disease images, further enhancements can improve its generalization to real-field conditions. Key directions for future work include:

- **Synthetic Data Augmentation:** Implement image-stitching and generative augmentation techniques to create composite images with multiple leaves and co-occurring diseases, enriching the training set with complex scenarios.

- **Multi-Disease Detection:** Extend the classification head to support multi-label outputs, enabling simultaneous detection of multiple diseases on a single leaf.

- **Lightweight Model Compression:** Explore quantization and pruning techniques to further reduce model size and latency for real-time inference on resource-constrained mobile devices.

- **Edge Deployment and User Interface:** Develop a user-friendly mobile application or edge-device integration, including offline inference capabilities and guided image capture, to facilitate adoption by farmers and extension workers.
- **Expanded Crop Coverage:** Incorporate additional crops and disease classes, leveraging transfer learning to scale the system's applicability across diverse agricultural contexts.
- **Field Validation Trials:** Conduct large-scale field trials to evaluate system efficacy, collect feedback, and calibrate the model for specific regional disease patterns.

11 CONCLUSION

In this report, a comprehensive AI-based plant disease detection system was developed using convolutional neural networks and transfer learning techniques. The individual crop models achieved validation accuracies ranging from 92.71% to 98.54%, while the consolidated model across 28 classes reached 95.15% validation accuracy. The comparative analysis demonstrated that transfer learning significantly outperforms training from scratch, both in accuracy and training efficiency. Real-world testing highlighted the system's robustness in controlled scenarios, though its performance decreased when presented with complex images containing multiple diseases or varied environmental conditions. Overall, the project validates the effectiveness of lightweight CNN architectures for mobile deployments and underscores the potential of AI in precision agriculture for early disease detection, potentially reducing crop losses and improving yield.

12 ACKNOWLEDGMENTS

Punjab AI Excellence is a program started by Dr. Sandeep Singh Sandha. Its main goal is to help students and young people in India learn about Artificial Intelligence (AI) and use it to solve real-life problems.

This program is known as one of the best in teaching AI. It focuses on how AI can be used to improve our daily lives, such as better healthcare, helping farmers with smart farming, improving education, and making government services faster and easier.

One special thing about Punjab AI Excellence is that it wants to turn villages into smart technology hubs. The goal is to support people in rural areas to master AI skills and use technology to create new jobs, help their communities, and build a better future. Punjab AI Excellence is helping students become future leaders in AI, making sure that even villages can lead in technology and innovation.

REFERENCES

Ahmad, M. H. (2025). Seasonal corn leaf disease dataset: A multi-year collection for robust analysis. Mendeley Data, V1.

Chollet, F. and the Keras Team (2023). Keras transfer learning and fine-tuning guide. https://keras.io/guides/transfer_learning/.

Government of India (2009). Rajya sabha question no. 1696: Loss of crop yield due to pest attacks. Answered by Prof. K.V. Thomas, Minister of State for Agriculture and Consumer Affairs, Food and Public Distribution.

Huyen, C. (2024). *AI Engineering*. O'Reilly Media, Inc.

Krishak Jagat (2023). Croplife india report: Inr 2 lakh crores of annual yield loss due to pests in india. https://www.agribusinessglobal.com/agrochemicals/croplife-india-report-inr-2-lakh-crores-of-annual-yield-loss-due-to-pests-in-india/.

Manvikar, A. and Reddy, P. (2023). Dataset of groundnut plant leaf images for classification and detection. Mendeley Data, V3.

Mohanty, S. P., Hughes, D. P., and Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Frontiers in Plant Science*, 7:1419.

PTI (2014). Crops worth rs 50,000 crore are lost a year due to pest, disease: Study. https://economictimes.indiatimes.com/news/economy/agriculture/crops-worth-rs-50000-crore-are-lost-a-year-due-to-pest-disease-study/articleshow/30345409.cms.

Rahman, M. S., Hasan, R., and Mojumdar, M. U. (2024). Mango leaf disease dataset. Mendeley Data, V1.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4510–4520.

Solapure, V., DY, S., and Jawale, A. (2024). Tomato leaf disease dataset. Mendeley Data, V1.

Sud, S. (2013). Crop loss at rs 1.5 lakh crore each year. https://www.business-standard.com/article/markets/crop-loss-at-rs-1-5-lakh-cr-each-yr-108021801055$_1.html.Lastupdated3 : 21AMIST$.

Tan, M. and Le, Q. V. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, volume 97, pages 6105–6114.