Deep Learning for Plant Disease Detection: A Practical Framework for Image-Based Crop Diagnosis

Debabrata Pruseth
AI Architect & Applied AI Researcher
Singapore

Author Note

This article is a research-style companion version of the author's blog post "Detecting Plant Diseases with AI – A Beginner-Friendly Deep Learning Project" and the associated GitHub project" Plant_disease_AI"

Abstract

Plant diseases can significantly affect crop productivity, food quality, and agricultural sustainability. Early identification of disease symptoms is therefore an important use case for computer vision and applied artificial intelligence in agriculture. This study presents a reproducible deep learning framework for multi-class plant disease classification using leaf images from the PlantVillage dataset. The work builds on the author's original implementation and compares a foundational EfficientNetB0 transfer learning baseline with an advanced EfficientNetB4-based model. The baseline model used ImageNet-pretrained EfficientNetB0, class weighting, sparse categorical cross-entropy, and partial fine-tuning of the final layers. It achieved 19.52% test accuracy, 39.69% top-3 test accuracy, and a macro F1-score of 0.0911, indicating limited performance across the 38 plant-disease classes. The advanced model used EfficientNetB4 with higher-resolution inputs, alpha-balanced categorical focal loss, mixed precision training, lightweight data augmentation, and full-backbone fine-tuning. The EfficientNetB4 model achieved a best validation accuracy of 99.59% and a best validation loss of 0.00494 after full fine-tuning. These results suggest that model capacity, image resolution, full-backbone adaptation, and class-imbalance-aware training can substantially improve plant disease classification performance on PlantVillage. However, because the current advanced model results are based on validation performance and the dataset largely consists of controlled leaf images, further test-set evaluation and external validation on field images are required before the framework can be considered deployment-ready.

Keywords: plant disease detection, deep learning, EfficientNet, transfer learning, PlantVillage, focal loss, Grad-CAM, agricultural AI, image classification, crop diagnosis.

---

1. Introduction

Plant disease detection is a critical task in agriculture because disease outbreaks can reduce yield, degrade crop quality, and increase the economic burden on farmers. In many agricultural settings, early disease symptoms appear visually on plant leaves through discoloration, spots, lesions, blight patterns, or texture changes. Traditional diagnosis depends on human observation by farmers, agronomists, or plant pathologists. Although expert diagnosis remains important, manual inspection can be time-consuming, difficult to scale, and inconsistent when expert support is unavailable.

Computer vision provides a practical opportunity to assist plant disease diagnosis by automatically classifying leaf images into healthy or diseased categories. The author's original blog presented this work as a beginner-friendly deep learning project for plant disease detection using transfer learning. The present manuscript reformulates that implementation into a research-style applied machine learning study. The goal is not to claim a new disease-detection algorithm, but to develop and evaluate a reproducible framework for image-based crop disease classification.

The study investigates the following research question:

Can an EfficientNet-based transfer learning pipeline provide an effective practical framework for multi-class plant disease classification from leaf images, and how does an advanced EfficientNetB4 configuration compare with a foundational EfficientNetB0 baseline?

To answer this question, the study compares two notebook-based experiments. The first uses EfficientNetB0 as a lightweight foundational baseline. The second upgrades the approach to EfficientNetB4 and introduces several improvements, including higher-resolution input, focal loss with alpha balancing, mixed precision training, data augmentation, and full fine-tuning.

The contribution of this work is therefore threefold. First, it provides a reproducible transfer learning pipeline for plant disease classification using the PlantVillage dataset. Second, it empirically compares a lightweight baseline model with a more advanced scaled model. Third, it identifies practical lessons about model capacity, fine-tuning strategy, and class imbalance in multi-class plant disease recognition.

---

2. Background and Related Work

Deep learning has become a widely used approach for image-based plant disease recognition. Leaf disease classification is commonly treated as a supervised image classification task in which a convolutional neural network learns disease-specific visual patterns from labelled images. Earlier studies demonstrated that deep learning can achieve strong performance on controlled leaf image datasets, while also noting that models trained on curated datasets may not generalize reliably to field conditions.

The PlantVillage dataset has become a common benchmark for this task. TensorFlow Datasets describes PlantVillage as containing 54,303 healthy and unhealthy leaf images across 38 classes. This makes it suitable for supervised multi-class classification experiments, although its controlled image conditions limit direct conclusions about real-world deployment.

EfficientNet was selected as the model family for this study because it provides a principled scaling approach for convolutional neural networks. EfficientNet uses compound scaling to balance network depth, width, and input resolution, allowing larger variants such as EfficientNetB4 to capture more detailed visual patterns than smaller variants such as EfficientNetB0. This is relevant for plant disease classification because disease symptoms can be visually subtle and may depend on fine-grained texture, lesion shape, or color differences.

Class imbalance is another important consideration. Some plant-disease categories contain many more examples than others, which can cause models to favour majority classes. The advanced experiment therefore uses focal loss, a loss function originally proposed to address class imbalance by reducing the influence of easy examples and focusing training on harder examples.

The project also includes Grad-CAM as an explainability method. Grad-CAM produces visual localization maps by using gradients flowing into the final convolutional layer, helping identify image regions that influence a CNN's prediction. In plant disease classification, this is useful because a model should ideally focus on disease-relevant leaf regions rather than background artifacts.

---

3. Dataset

The study uses the **PlantVillage** dataset through TensorFlow Datasets. The dataset contains leaf images representing healthy and diseased plant conditions across multiple crop types and disease categories.

The experimental setup uses a custom split:

Split	Ratio
Training	80%
Validation	10%
Test	10%

The notebooks use the following split structure:

```
split=['train[:80%]', 'train[80%:90%]', 'train[90%:]']
```

The dataset contains **38 classes**, and the original images are represented as RGB images. The B0 and B4 experiments differ in input image size. EfficientNetB0 uses 224 × 224 inputs, while EfficientNetB4 uses 380 × 380 inputs.

Dataset property	Value
Dataset	PlantVillage
Source	TensorFlow Datasets
Number of classes	38
Original image shape	256 × 256 × 3
B0 input size	224 × 224 × 3
B4 input size	380 × 380 × 3
Training split	80%
Validation split	10%
Test split	10%
Batch size	32
Random seed	42

The dataset is suitable for controlled experimentation, but the images are not fully representative of real-world field conditions. Therefore, performance on PlantVillage should be interpreted as benchmark performance rather than proof of field readiness.

---

4. Methodology

4.1 Problem Formulation

The task is formulated as supervised multi-class classification. Given an input leaf image x, the model predicts a probability distribution over 38 plant-disease categories. The predicted class is selected as the category with the highest predicted probability.

4.2 Foundational Baseline: EfficientNetB0

The foundational notebook implements a baseline model using **EfficientNetB0** with ImageNet-pretrained weights. The original classification layer is removed, and a custom classification head is added for the 38 PlantVillage classes.

The B0 experiment uses:

Component	Configuration
Backbone	EfficientNetB0
Pretraining	ImageNet
Input size	224 × 224 × 3
Loss	Sparse categorical cross-entropy
Optimizer	Adam
Stage 1 learning rate	1e-3
Stage 2 learning rate	1e-5
Stage 1 epochs	8
Stage 2 epochs	6
Fine-tuning strategy	Last 50 layers unfrozen
Class imbalance handling	Class weights
Metrics	Accuracy, top-3 accuracy, precision, recall, F1
Evaluation	Test-set classification report and confusion matrix

The training process follows a two-stage transfer learning strategy. In Stage 1, the EfficientNetB0 base model is frozen and only the classification head is trained. In Stage 2, the final 50 layers of the base model are unfrozen and fine-tuned with a lower learning rate.

This baseline provides a useful reference point because it represents a lightweight transfer learning approach that is easier to train and computationally less demanding than the larger EfficientNetB4 model.

4.3 Advanced Model: EfficientNetB4

The advanced notebook implements a larger and more refined model using **EfficientNetB4** with ImageNet-pretrained weights. EfficientNetB4 is selected because it has greater representational capacity and uses a higher input resolution than EfficientNetB0.

The advanced architecture is:

Layer / Component	Description
Input	380 × 380 × 3

Layer / Component	Description
Backbone	EfficientNetB4, ImageNet weights, `include_top=False`
Pooling	GlobalAveragePooling2D
Dropout 1	0.2
Dense layer	256 units, ReLU activation
Dropout 2	0.3
Output layer	Dense layer with 38 outputs
Final activation	Softmax with float32 output

The final softmax activation is forced to float32 to support numerical stability under mixed precision training. This is a useful implementation detail because the notebook enables mixed precision globally.

The EfficientNetB4 experiment also uses a two-stage strategy:

Stage	Backbone status	Learning rate	Loss	Epochs
Stage 1	Frozen	1e-4	Alpha-balanced categorical focal loss	25
Stage 2	Fully unfrozen	1e-5	Alpha-balanced categorical focal loss	25

The Stage 1 experiment trains only the classification head. In Stage 2, the full EfficientNetB4 backbone is unfrozen and fine-tuned with a lower learning rate.

4.4 Data Preprocessing and Augmentation

The B4 notebook uses the following preprocessing:

Step	Description
Resize	Images resized to 380 × 380
Normalization	Pixel values scaled to [0, 1]

Step	Description
Label encoding	Labels converted to one-hot vectors
Batch size	32
Prefetching	Used through TensorFlow data pipeline

For training images, the notebook applies lightweight augmentation:

Augmentation	Purpose
Random horizontal flip	Improves robustness to leaf orientation
Random brightness	Simulates lighting variation
Random contrast	Simulates visual contrast variation

Validation and test images are resized and normalized but not augmented.

4.5 Class Imbalance Handling

The B0 notebook uses class weights with sparse categorical cross-entropy. The B4 notebook uses a more advanced approach: **alpha-balanced categorical focal loss**.

The B4 notebook computes class weights using the training distribution and incorporates those values into the focal loss alpha term. The focal loss uses:

Parameter	Value
Alpha	Computed from class weights
Gamma	2.0

This approach is intended to address two related problems. First, alpha balancing gives higher weight to underrepresented classes. Second, focal loss reduces the contribution of easy examples and focuses training on harder or misclassified examples. This is especially relevant for PlantVillage because some classes are much more frequent than others.

The class weights in the notebooks indicate clear class imbalance. For example, the B0 notebook reports early class weights such as:

Class index	Weight
Class 00	2.2819
Class 01	2.4480
Class 02	5.1729
Class 03	0.8754
Class 04	0.9448

This distribution supports the decision to include class-imbalance-aware training.

4.6 Mixed Precision Training

The B4 notebook enables mixed precision training using:

```
mixed_precision.set_global_policy('mixed_float16')
```

The optimizer is wrapped with a loss-scaling optimizer:

```
mixed_precision.LossScaleOptimizer
```

Mixed precision can improve GPU training efficiency by using lower precision arithmetic where appropriate, while loss scaling helps maintain numerical stability. The final softmax activation is set to float32, which is a practical safeguard for stable probability outputs.

4.7 Explainability with Grad-CAM

The broader project includes Grad-CAM as an interpretability method. Grad-CAM can be used to visualize which image regions contributed most strongly to the model's prediction. In the context of plant disease detection, this can help assess whether the model is focusing on disease-relevant regions such as leaf spots, lesions, discoloration, or blight patterns.

However, Grad-CAM should be interpreted as a diagnostic aid rather than proof of causal reasoning. It can show where the model is attending, but it does not prove that the model has learned biological disease mechanisms.

---

5. Experimental Setup

The experiments were implemented in Jupyter/Google Colab using TensorFlow and TensorFlow Datasets. The B4 notebook explicitly states that it was run in Google Colab using GPU acceleration.

The two main experiments are summarized below.

Configuration	EfficientNetB0 Baseline	EfficientNetB4 Advanced
Dataset	PlantVillage	PlantVillage
Classes	38	38
Input size	224 × 224	380 × 380
Backbone	EfficientNetB0	EfficientNetB4
Pretraining	ImageNet	ImageNet
Loss	Sparse categorical cross-entropy	Alpha-balanced categorical focal loss
Imbalance handling	Class weights	Class weights incorporated as focal loss alpha
Optimizer	Adam	Adam with mixed precision loss scaling
Stage 1 LR	1e-3	1e-4
Stage 2 LR	1e-5	1e-5
Fine-tuning	Last 50 layers	Full backbone
Stage 1 epochs	8	25
Stage 2 epochs	6	25
Evaluation	Test set	Validation shown; test evaluation pending
Additional methods	Confusion matrix, classification report	Training curves; Grad-CAM planned/available in broader project

---

6. Results

6.1 EfficientNetB0 Baseline Results

The EfficientNetB0 baseline performed poorly on the 38-class PlantVillage classification task. The best validation accuracy reached during training was 19.46%. On the test set, the model achieved 19.52% accuracy and 39.69% top-3 accuracy.

Metric	EfficientNetB0 Result
Best validation accuracy	19.46%
Test accuracy	19.52%
Test top-3 accuracy	39.69%
Macro F1-score	0.0911
Micro F1-score	0.1952
Macro precision	0.1518
Macro recall	0.1346
Misclassified test samples	4,370 out of 5,430

The low macro F1-score suggests that the model did not perform consistently across all classes. This is important because overall accuracy alone can hide poor performance on minority or visually similar classes. The B0 model therefore serves as a useful baseline demonstrating that a lightweight transfer learning model with limited fine-tuning may be insufficient for this multi-class plant disease classification problem.

6.2 EfficientNetB4 Stage 1 Results: Frozen Backbone

The EfficientNetB4 model first trained only the classification head while the backbone remained frozen. This stage showed limited learning.

Stage	Training accuracy	Training loss	Validation accuracy	Validatio n loss
B4 Stage 1 final epoch	8.70%	3.4052	10.94%	3.3902
B4 Stage 1 best validation accuracy	—	—	11.64%	—

These results indicate that, for this setup, training only the classification head was not sufficient. The frozen EfficientNetB4 features did not directly transfer well enough to classify the 38 PlantVillage categories without deeper adaptation.

This is an important experimental observation: simply attaching a classifier head to a pretrained model may not be adequate for fine-grained agricultural image classification.

6.3 EfficientNetB4 Stage 2 Results: Full Fine-Tuning

In Stage 2, the full EfficientNetB4 backbone was unfrozen and fine-tuned with a lower learning rate of 1e-5. This produced a dramatic improvement.

Metric	EfficientNetB4 Stage 2 Result
Best validation accuracy	99.59%
Best validation loss	0.00494
Final training accuracy	99.55%
Final training loss	0.0046
Final validation accuracy	99.17%
Final validation loss	0.0174

The best validation result was observed at Epoch 23, where the model achieved 99.59% validation accuracy and 0.00494 validation loss.

The results show that full fine-tuning was the major performance driver in the advanced experiment. While the frozen-backbone stage achieved only around 11% validation accuracy, full fine-tuning rapidly improved performance to above 99% validation accuracy.

6.4 Comparative Results

The comparison between the B0 baseline and B4 advanced model is summarized below.

Experiment	Backbone	Input Size	Loss	Fine-tuning Strategy	Best Validation Accuracy	Test Accuracy	Macro F1
Baseline	EfficientNetB0	224 × 224	Sparse categorical cross-entropy + class weights	Last 50 layers	19.46%	19.52%	0.0911
Advanced Stage 1	EfficientNetB4	380 × 380	Alpha-balanced focal loss	Frozen backbone	11.64%	Not evaluated	Not evaluated

Experiment	Backbone	Input Size	Loss	Fine-tuning Strategy	Best Validation Accuracy	Test Accuracy	Macro F1
Advanced Stage 2	EfficientNetB4	380 × 380	Alpha-balanced focal loss	Full backbone	99.59%	Pending	Pending

The results suggest that the improved performance cannot be attributed to a single factor alone. The B4 experiment differs from the B0 baseline in several ways, including model capacity, input resolution, loss function, number of epochs, label encoding, mixed precision, augmentation, and fine-tuning depth. Therefore, the improvement should be interpreted as the result of an enhanced training configuration rather than as evidence that EfficientNetB4 alone is responsible for the performance gain.

---

7. Discussion

The experiments provide several practical insights.

First, the EfficientNetB0 baseline demonstrates the difficulty of multi-class plant disease classification when using a lightweight model and limited fine-tuning. The model's low test accuracy and macro F1-score indicate that it struggled to learn discriminative features across all 38 classes. This may be due to the complexity of the task, visual similarity between disease classes, class imbalance, limited model capacity, or suboptimal preprocessing and fine-tuning choices.

Second, the EfficientNetB4 frozen-backbone result shows that pretrained features alone were insufficient in this case. Although EfficientNetB4 is more powerful than EfficientNetB0, the frozen feature extractor did not achieve strong classification performance when only the head was trained. This suggests that plant disease recognition may require deeper adaptation of the backbone to capture domain-specific visual features.

Third, full fine-tuning of EfficientNetB4 produced a substantial improvement. Once the full backbone was unfrozen, validation accuracy increased rapidly, reaching 99.59%. This suggests that the model benefited from adapting its convolutional representations to the specific visual characteristics of plant leaves and disease symptoms.

Fourth, class-imbalance-aware training appears methodologically appropriate for this dataset. The use of alpha-balanced focal loss in the B4 experiment is well aligned with the observed imbalance in class weights. However, to isolate the effect of focal loss, a future ablation study should compare EfficientNetB4 trained with standard cross-entropy against EfficientNetB4 trained with focal loss.

Fifth, the large performance gap between B0 and B4 must be interpreted cautiously. The B4 model differs from the B0 model across multiple experimental dimensions. Therefore, the current results demonstrate the effectiveness of the advanced pipeline as a whole, but they do not isolate the individual contribution of each design choice.

---

8. Limitations and Threats to Validity

This study has several limitations.

First, the B4 advanced model has not yet been reported on the held-out test set. The validation accuracy is very high, but a complete research paper should include test accuracy, macro F1, weighted F1, per-class precision and recall, and confusion matrix results for the B4 model.

Second, the PlantVillage dataset contains controlled leaf images. Real-world farm images may include complex backgrounds, varying lighting, shadows, multiple leaves, soil, occlusion, blur, different camera types, and disease symptoms at different severity stages. Therefore, high PlantVillage performance should not be interpreted as proof of real-world diagnostic readiness.

Third, the comparison between EfficientNetB0 and EfficientNetB4 is not a controlled ablation. The experiments differ in model size, image resolution, training length, loss function, augmentation, label encoding, and fine-tuning strategy. Future work should isolate these factors through controlled experiments.

Fourth, disease severity is not estimated. The model predicts disease class, but it does not quantify the severity of infection or recommend treatment actions.

Fifth, Grad-CAM provides useful visual explanations but does not prove that the model has learned plant-pathology-grounded reasoning. Visual explanations should therefore be used as a supporting diagnostic tool, not as definitive evidence of model reliability.

---

9. Future Work

Future work should first complete the B4 evaluation on the held-out test set. This should include accuracy, macro F1, weighted F1, precision, recall, top-3 accuracy, confusion matrix, and per-class performance.

Second, an ablation study should be conducted to isolate the effect of each improvement. Useful comparisons would include:

Experiment	Purpose
B4 + cross-entropy	Test impact of architecture alone
B4 + focal loss	Test impact of focal loss

Experiment	Purpose
B4 frozen backbone	Test transfer learning without fine-tuning
B4 partial fine-tuning	Compare partial versus full fine-tuning
B4 without augmentation	Test augmentation contribution
B4 with field images	Test real-world robustness

Third, the model should be evaluated on external field-captured leaf images. This is necessary to assess generalization beyond controlled datasets.

Fourth, Grad-CAM visualizations should be systematically analysed to determine whether the model attends to disease-relevant leaf regions. Misclassified examples should be reviewed to understand whether errors are caused by visually similar diseases, class imbalance, background artifacts, or ambiguous symptoms.

Fifth, the framework could be extended into a hierarchical classifier. The first stage could classify plant species, and the second stage could classify disease within that plant species. This may reduce confusion between disease classes across different crops.

Finally, future versions could evaluate newer architectures such as EfficientNetV2, ConvNeXt, or Vision Transformers, while also considering deployment constraints for mobile or edge-based agricultural applications.

---

10. Conclusion

This study presents a reproducible EfficientNet-based transfer learning framework for multi-class plant disease classification using the PlantVillage dataset. The foundational EfficientNetB0 baseline achieved limited performance, with 19.52% test accuracy, 39.69% top-3 accuracy, and a macro F1-score of 0.0911. This result highlights the difficulty of the 38-class classification task when using a lightweight model with limited fine-tuning.

The advanced EfficientNetB4 model substantially improved validation performance after full fine-tuning, achieving a best validation accuracy of 99.59% and a best validation loss of 0.00494. The results suggest that higher model capacity, higher input resolution, full-backbone adaptation, alpha-balanced focal loss, mixed precision training, and data augmentation together form a much stronger pipeline for PlantVillage classification.

However, the study should be interpreted as an applied benchmark and reproducible framework rather than a deployment-ready crop diagnosis system. The B4 model still requires held-out test-set evaluation, and the overall framework requires external validation on real-world field images. With these additions, the work can be strengthened into a credible applied machine learning manuscript suitable for sharing as a technical report, preprint, or portfolio-based research contribution.

Suggested Citation

Pruseth, D. (2025). *Deep Learning for Plant Disease Detection: A Practical Framework for Image-Based Crop Diagnosis.*