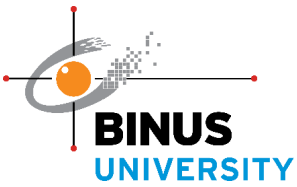
Potato Leaf Disease Classification Using Deep Learning: A Comparative Study of Custom CNN and Transfer Learning Approaches

Final Report

By :

Mikhael Henokh Santoso	2702239750
Frederick Krisna Suryopranoto	2702272382
Dustin Manuel	2702296016



Deep Learning - COMP6826001

5th Semester

Computer Science

School of Computer Science

UNIVERSITAS BINA NUSANTARA

2025

Abstract

This project has the objective to develop an automated system to classify potato leaf disease using deep learning. Two approaches were implemented and compared: a custom Convolutional Neural Network (CNN) and a transfer learning model based on ResNet50. The system classifies potato leaves into three categories - Healthy leaves, Early Blight, and Late Blight. The complete pipeline includes data preprocessing, model training , evaluation, and deployment into a Streamlit web-application for real-time predictions. The ResNet50 model achieved 98.77% in accuracy, while custom CNN achieved 99.26% in accuracy. The deployed application enhances the accessibility of this technology for practical agriculture deployment. This research shows the capability of deep learning to effectively address critical, real-world challenges in agriculture and crop management.

Keywords : deep learning, CNN, transfer learning, plant disease, classification, computer vision, agriculture

Chapter 1. Introduction

1.1 Background

Plant diseases remain a major challenge for farmers around the world, as they can lead to significant crop losses every year. Detecting these diseases early and accurately is crucial, yet still difficult to achieve in practice. Traditional methods such as manual inspection by agricultural experts are often slow, costly, and inconsistent. Potato plants, in particular, are highly vulnerable to severe diseases like Early Blight and Late Blight, which can destroy an entire harvest if not identified and addressed promptly.

1.2 Problem Statement

Farmers encounter several persistent challenges in plant disease management :

- Limited access to experts who specialize in disease diagnosis.
- Field inspections are often labor-intensive, time-consuming, and expensive.
- Human assessments or judgments can also be inconsistent
- Early symptoms of disease are frequently difficult to detect with the naked eye
- Existing conventional detection methods lacks the scalability required for modern large-scale agriculture

1.3 Objectives

The primary objectives of this project are:

- To develop an automated system for identifying potato leaf diseases from digital images.
- To compare and optimize the performance of two deep learning architectures: a Custom CNN and a pre-trained ResNet50.
- To deploy a user-friendly web application that democratizes access to expert-level diagnosis.

1.4 Significance

This study contributes to both agriculture and technology by:

- Enhancing Food Security: Enabling early, accurate detection to minimize crop losses.
- Accessibility: Providing farmers with an accessible, expert-level diagnostic tool without requiring specialized knowledge.
- Scalability: Establishing a practical Deep Learning framework that can be adapted for other crops and diseases.

Chapter 2. Dataset

2.1 Dataset Composition and Description

The study utilizes a publicly available dataset comprising 4,072 labeled potato leaf images. To ensure model robustness, the dataset includes variations in lighting, background, and leaf orientation. The images are categorized into three distinct classes:

- Early Blight (1,628 images): Represents early-stage fungal infections characterized by dark spots with concentric rings.
- Late Blight (1,424 images): Depicts aggressive infections caused by *Phytophthora infestans*, showing irregular water-soaked lesions.
- Healthy (1,020 images): Disease-free leaves exhibiting normal texture and coloration.

2.2 Data Partitioning

To facilitate proper model training, hyperparameter tuning, and unbiased evaluation, the dataset was split into three subsets:

- Training Set (3,251 images): Used for learning features and weights.
- Validation Set (416 images): Used for monitoring performance and preventing overfitting during training.
- Test Set (405 images): Reserved strictly for the final performance assessment on unseen data.

Chapter 3. Modeling

3.1 Data Preprocessing

All of the images used were resized to dimensions of 224x224 pixels. To enhance the model generalization and robustness during the training, there were several augmentation techniques applied : Random horizontal flips, Random rotation up to 30 degrees, Color adjustment (brightness, contrast, saturation), and Normalization using standard ImageNet statistics. For the validation and testing subset, only resizing and normalization were executed.

3.2 Model Architecture

3.2.1 Custom CNN

We engineered a custom built CNN with 4 sequential convolutional blocks :

- Input Layer: (224×224×3 RGB images)
- Convolutional Block 1:
  Conv2D (32 filters, 3×3 kernel, stride=1, padding=1), BatchNorm2D, ReLU activation, MaxPool2D (2×2), stride=2, Output (112×112×32)
- Convolutional Block 2:
  Conv2D (64 filters, 3×3 kernel, stride=1, padding=1), BatchNorm2D,ReLU activation, MaxPool2D (2×2), stride=2, Output (56×56×64)
- Convolutional Block 3:
  Conv2D (128 filters, 3×3 kernel, stride=1, padding=1), BatchNorm2D, ReLU activation, MaxPool2D (2×2), stride=2, Output (28×28×128)
- Convolutional Block 4:
  Conv2D (256 filters, 3×3 kernel, stride=1, padding=1), BatchNorm2D, ReLU activation, MaxPool2D (2×2), stride=2, Output (14×14×256)
- Flatten: (14×14×256) = 50,176 features
- Fully Connected Block:

Linear (50,176 → 512), ReLU activation, Dropout (p=0.5), Linear (512 → 256), ReLU activation, Dropout (p=0.5)

- Output Layer:
  Linear (256 → 3) (number of classes), Softmax (applied during inference)

3.2.2 ResNet50 Transfer Learning

Base Model: ResNet50 pretrained on ImageNet

- Frozen layers: First 47 layers (feature extraction)
- Fine-tunable layers: Last 3 layers + classifier

Modified Classifier:

- AdaptiveAvgPool2D: 2048 features
- Linear: 2048 → 512
- ReLU activation
- Dropout (p=0.5)
- Linear: 512 → 3 (number of classes)

3.3 Training Setup

- Loss Function : CrossEntropyLoss
- Optimizer : Adam, configured with initial learning rate 0.001
- Early Stopping : Training will be halted if the validation fail to improve for 5 consecutive epochs
- Batch Size : 32
- Epochs : 50 epochs

Chapter 4. Evaluation

4.1 Evaluation Metrics

- Accuracy : The overall proportion of correct classifications
- Precision : The ratio of true positive predictions to all positive predictions made
- Recall : The ratio of true positive predictions to all actual positive instances
- F1-Score : The harmonic mean of precision and recall, providing a balanced measure
- Confusion Matrix : A visualization detailing the exact distribution of correct and incorrect classifications across all classes

- Inference Time : The computational latency required to generate a single prediction

4.2 System Implementation

The complete system architecture was structured into modular components :

1. Data Pipeline (data.py) : Responsible for image loading, preprocessing, and creation for train-validation-test splits
2. Model Definitions (model.py) : Contains the code definition for both custom CNN and ResNet50 architectures
3. Training Script (train.py) : Manages the training loop, model checkpoints, and performance logging
4. Evaluation (evaluate.py) : Computes all performance metrics and generates the necessary visualizations
5. Web App (app.py) : A Streamlit-based interface allowing users to upload images and receive real-time predictions

4.3 Experimental Results

4.3.1 Training Performance

Custom CNN :

- Best validation Accuracy : 98.80%
- Best Validation Loss: 0.0365
- Convergence epoch : 38

ResNet50 :

- Best validation accuracy : 99.52%
- Best Validation Loss: 0.0109
- Convergence epoch : 11

Both model shows good training dynamics, with minimal evidence of severe overfitting

4.3.2 Test Set performance

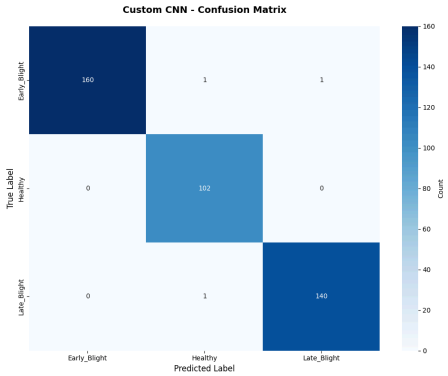
Metrics	Custom CNN	ResNet50
Accuracy	99.26%	98.77%
Precision (Avg)	99.27%	98.78%

Recall (Avg)	99.26%	98.77%
F1-Score (Avg)	99.26%	98.77%
Inference time	9.51 Second	8.83 Second
Model Size	26.49 MB	93.68 MB

4.3.3 Confusion Matrix Analysis

The primary source of misclassification involved in confusion between Early Blight and Late Blight , this is understandable due to the similarity of both diseases, but importantly, healthy leaves are rarely misclassified as diseased.

Custom CNN :



Interpretation :

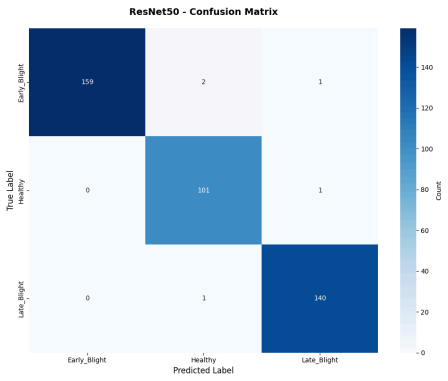
Excellent Early Blight recognition : 160/162 correct → 98.76% accuracy

Healthy leaves recognition : 102/102 correct -> 100% accuracy

Late Blight detection : 140/141 correct → 99.29% accuracy

Very minimal cross-class confusion **:** Only 3 misclassifications across all classes.

ResNet 50 :



Interpretation :

Early Blight recognition : 159/162 correct → 98.14% accuracy.

Healthy leaves recognition : 101/102 correct -> 99.02% accuracy

Late Blight detection : 140/141 correct → 99.29% accuracy.

Very minimal cross-class confusion : Only 5 misclassifications across all classes

4.3.4 Visualizations

Several plots are generated :

- Training loss and accuracy progression for both models
- Confusion matrices illustrating detailed prediction patterns

- Representative sample predictions annotated with confidence scores
- Comparative chart summarizing key model performance metrics

Chapter 5. Deployment

5.1 Web Application

The Streamlit-based application provides the following functional capabilities :

- Intuitive interface for a simple image upload
- Real-time prediction display, including confidence percentages
- Accessible information and description for each diagnosed disease
- Option for the user to select between Custom CNN or ResNet50 model for prediction
- Reference sample image for user orientation

To facilitate broad accessibility, the application is hosted on Streamlit Cloud. This allows users to access the diagnostic tool remotely via a web URL, requiring an active internet connection but eliminating the need for local computational resources.

Chapter 6. Reflection and Conclusion

6.1 Critical Analysis of Results

The project yielded results that challenged initial theoretical assumptions. Contrary to the expectation that Transfer Learning (ResNet50) would dominate due to its pre-training on massive datasets, the **Custom CNN** emerged as the superior model. It achieved higher accuracy (**99.26%** vs. 98.77%) and significantly better efficiency in terms of storage (**26.46MB** vs. 93.68MB). While ResNet50 maintained a slight edge in inference speed (21.81 ms), the trade-off favors the Custom CNN for environments where storage is a constraint.

These results were achieved despite several technical challenges, including hardware limitations that restricted batch sizes and a relatively modest dataset scale of 4,072 images with slight class imbalances. This demonstrates that for specific, narrow-domain tasks like potato leaf disease detection, a well-tuned lightweight model can often outperform complex generalized architectures.

6.2 System Limitations and Practical Implications

While the model achieves high accuracy, its operational boundary is defined by several limitations:

- **Scope & Input Sensitivity:** The system is strictly limited to potato leaves and requires high-quality, close-up images. It lacks robustness against blurry photos, distant shots, or whole-plant photography.
- **Diagnostic Depth:** The current classification is categorical. It does not provide granularity regarding infection severity or spatial localization of the disease on the leaf.
- **Operational Requirements:** As a web-based tool, it requires users to possess modern computing devices and a stable internet connection for the hosted application. Consequently, the system functions as an assistive diagnostic tool rather than a replacement for professional agricultural consultation.

6.3 Key Project Learning

The development process provided the team with critical insights beyond technical coding skills:

- **Model Unpredictability:** Theoretical superiority does not always translate to practical performance; empirical testing is crucial.
- **Data-Centric AI:** The quality and preprocessing of the dataset proved more impactful on the final accuracy than the complexity of the model architecture.
- **Deployment Gap:** Moving a model from a local Jupyter Notebook to a production-ready Web Application (Streamlit) introduces unique challenges regarding environment consistency and latency.
- **Resource Management:** Hardware constraints act as a creative limit, forcing optimization in batch sizing and model complexity.

6.4 Conclusion and Future Roadmap

In conclusion, the study successfully established a highly accurate (99.26%) disease classification system for potato leaves. To evolve this prototype into a robust agricultural product, future work will focus on three key areas:

1. **Mobile Development:** creating a native mobile application to enable offline usage for farmers in remote fields.

2. **Predictive Modeling:** integrating external weather data to forecast disease outbreaks before they occur visually.
3. **Scope Expansion:** enlarging the dataset to cover a wider variety of crops and diseases, thereby increasing the tool's utility for a broader user base.

Chapter 7. References

[1] R. Mahum et al., "A novel framework for potato leaf disease detection using an efficient deep learning model," Human and Ecological Risk Assessment: An International Journal, vol. 29, no. 2, pp. 303-326, 2023. [Online]. Available: https://doi.org/10.1080/10807039.2022.2064814

[2] D. Tiwari, M. Ashish, N. Gangwar, A. Sharma, S. Patel, and S. Bhardwaj, "Potato Leaf Diseases Detection Using Deep Learning," in 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2020, pp. 461-466. [Online]. Available: https://doi.org/10.1109/ICICCS48265.2020.9121067

[3] S. Bangari, P. Rachana, N. Gupta, P. S. Sudi, and K. K. Baniya, "A Survey on Disease Detection of a potato Leaf Using CNN," in 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS), Coimbatore, India, 2022, pp. 144-149. [Online]. Available: https://doi.org/10.1109/ICAIS53314.2022.9742963

[4] M. Sardogan, A. Tuncer, and Y. Ozen, "Plant Leaf Disease Detection and Classification Based on CNN with LVQ Algorithm," in 2018 3rd International Conference on Computer Science and Engineering (UBMK), Sarajevo, Bosnia and Herzegovina, 2018, pp. 382-385. [Online]. Available: https://doi.org/10.1109/UBMK.2018.8566635

[5] C. Sarkar, D. Gupta, U. Gupta, and B. B. Hazarika, "Leaf disease detection using machine learning and deep learning: Review and challenges," Applied Soft Computing, vol. 145, p. 110534, 2023. [Online]. Available: https://doi.org/10.1016/j.asoc.2023.110534

[6] M. E. H. Chowdhury et al., "Automatic and Reliable Leaf Disease Detection Using Deep Learning Techniques," AgriEngineering, vol. 3, no. 2, pp. 294-312, 2021. [Online]. Available: https://doi.org/10.3390/agriengineering3020020

Chapter 8. Appendix

8.1 Team Contribution

Member	Contribution
Dustin Manuel	Web application making, Evaluation, making final report, website testing, making PPT
Frederick Krisna Suryopranoto	Data collecting, training ResNet50, Streamlit deployment, web testing, making GitHub repo
Mikhael Henokh Santoso	Data preprocessing, training custom CNN, visualization, website testing, contribute in GitHub repo, making final report

GitHub Repository: https://github.com/NSquid/Deep-Learning-AOL

Deployed app: https://potatoes-disease-classifications.streamlit.app/

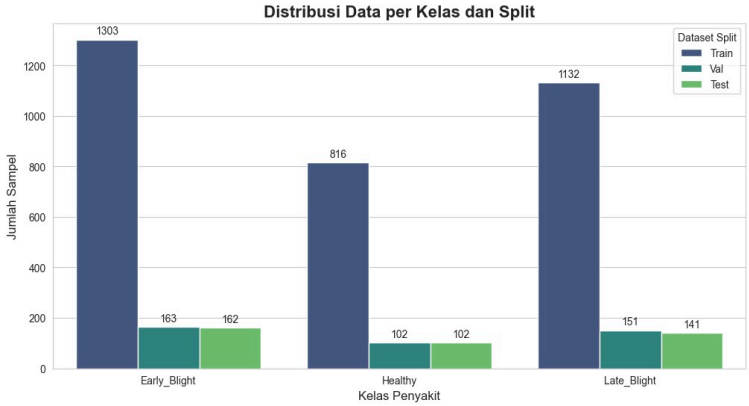
Presentation Slides:

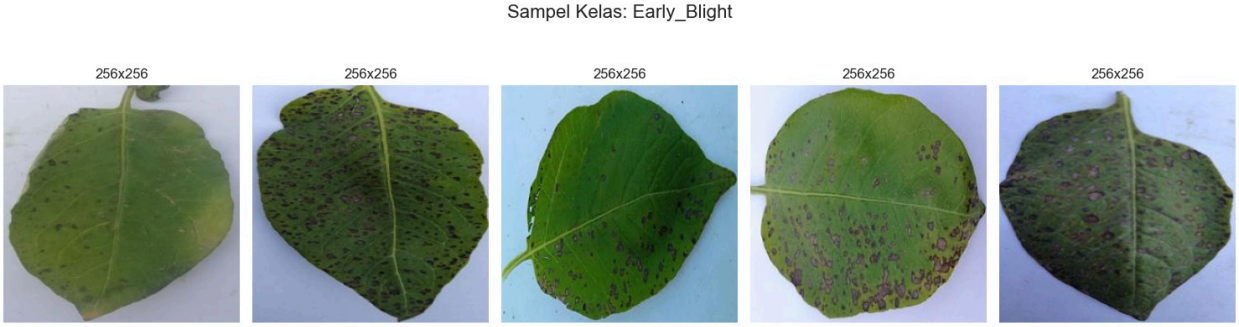
https://www.canva.com/design/DAG7GJiBW3o/9eeT19_0ew4cT5dRQFP-5Q/edit?utm_content=DAG7GJiBW3o&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton

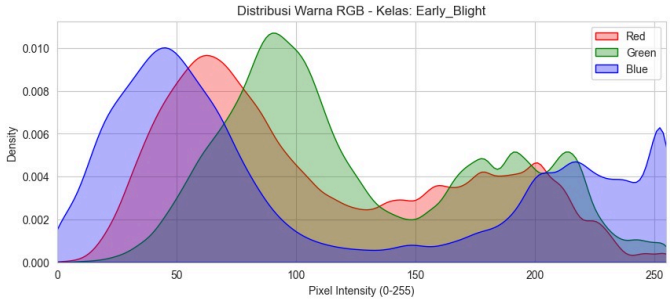
Demo Video:

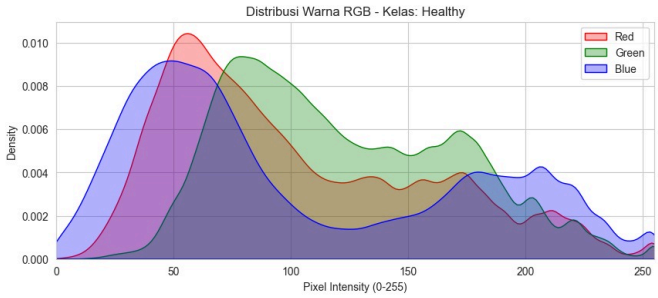
https://drive.google.com/file/d/1e5LJiWHrVBlMoY6yfqvmAjIWVcsTzjtO/view?usp=sharing

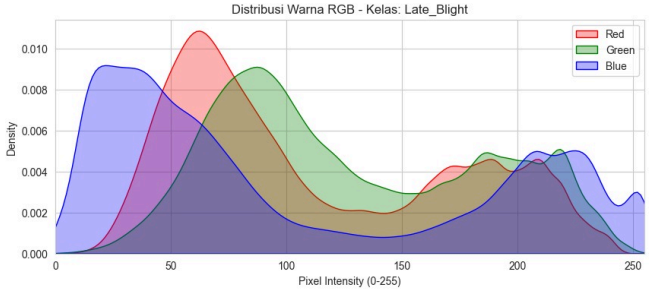
8.2 Additional Screenshots

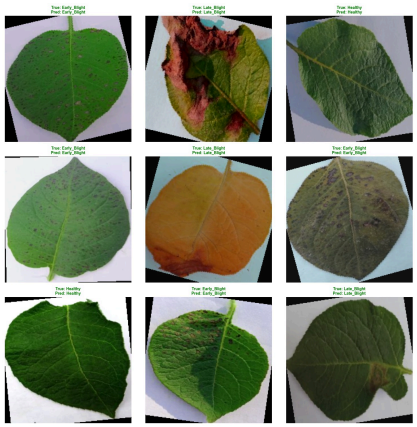












Simulasi Augmentasi Data (Resize, Rotate, Flip, Jitter)