Computer Vision
Project Report
Agricultural Pest Detection

June 14, 2025

Abstract

This report presents the design, development, and evaluation of an intelligent computer vision system for crop pest detection using the Agricultural Pests Image Dataset from Kaggle. The system aims to reduce the burden of manual pest monitoring by leveraging state-of-the-art deep learning models, namely ResNet18 and EfficientNetB0, trained to classify 12 pest categories. The proposed solution addresses critical challenges in agricultural automation and demonstrates deployment feasibility through a prototype implementation.

1 Introduction

The agricultural sector is confronted with significant threats posed by pest infestations, which have a profound effect on crop yield and food security. Timely and precise detection of these pests is essential for effective pest management. Conventional methods for pest identification are often labour-intensive and prone to errors, especially in extensive farming settings.

This project investigates a solution based on deep learning for the automated classification of pests using image data. We employ transfer learning with pre-trained models such as ResNet and EfficientNet, fine-tuning them to achieve optimal performance on the Agricultural Pests Image Dataset sourced from Kaggle. To enhance model performance, we implement transfer learning for adaptive hyperparameter tuning and explore EfficientNet variants (e.g., B2–B5). This approach is designed to maximize classification accuracy while ensuring computational efficiency.

2 Objectives

- Construct a strong image classification model aimed at pest detection.
- Employ EfficientNet and ResNet with fine-tuning techniques for transfer learning.
- Adjust hyperparameters dynamically through transfer learning to enhance model performance.

- Test various EfficientNet variants to assess the trade-offs between depth and accuracy.
- Attain a test accuracy of 94% or higher across several models.
- Assess and contrast the performance of 6 to 7 models on the specified dataset.
- Guarantee the scalability and generalizability of the solution for practical deployment.

3 Methodology

3.1 Dataset

- **Name:** Agricultural Pests Image Dataset
- **Description:** Contains labelled pest images from 12 categories, captured in various agricultural environments.
- **Preprocessing:** Resizing, normalization, augmentation (flips, zoom, rotations, cropping, color jitter)

3.2 Models and Techniques

- **Transfer Learning Models:** ResNet, EfficientNet (including B0 to B5)
- **Fine-Tuning:**
  - Unfreezing deeper layers selectively
  - Adding custom classification heads
- **Hyperparameter Optimization:**
  - Grid search over learning rate, batch size, dropout, and layer freeze levels
- **Tools and Libraries:** Python, PyTorch, torchvision, scikit-learn, matplotlib, seaborn

3.3 Experimental Process

1. Preprocess and augment dataset
2. Load pre-trained models and customize final layers
3. Perform grid search for each model to identify optimal hyperparameters
4. Train and validate models using stratified data splits (train/val/test)
5. Evaluate test accuracy, confusion matrix, and training loss curves

4 Evaluation Results and Visual Examples

4.1 Model Performance

Both models demonstrated high classification accuracy. EfficientNetB0 slightly outperformed ResNet18 in both validation and test phases.

- **Best Validation Accuracy:** 92.48%
- **Test Accuracy:** 94.55%
- **Training Time:** 15 minutes, 32 seconds

--- Classification Report ---

	precision	recall	f1-score	support
ants	0.95	0.96	0.96	57
bees	0.96	0.90	0.93	59
beetle	0.92	0.76	0.83	63
catterpillar	0.94	0.85	0.89	74
earthworms	0.90	0.90	0.90	49
earwig	0.79	0.94	0.86	67
grasshopper	0.95	0.92	0.93	84
moth	0.99	0.98	0.98	95
slug	0.88	0.96	0.92	54
snail	1.00	0.99	0.99	76
wasp	0.91	0.99	0.95	74
weevil	0.96	0.99	0.97	73

Figure 1: Classification Report

4.2 Training Logs

Epoch	Train Loss (Accuracy)	Val Loss (Accuracy)
38	0.2726 (0.9118)	0.2818 (0.9175)
39	0.2642 (0.9163)	0.2928 (0.9150)
40	0.2609 (0.9165)	0.2720 (0.9199)

Table 1: Training and Validation Performance

4.3 Classification Metrics

Class	Precision	Recall	F1-Score	Support
ants	0.90	0.97	0.93	66
bees	0.94	0.95	0.95	82
beetle	0.84	0.68	0.75	66
grasshopper	0.92	0.91	0.91	70
caterpillar	0.95	0.96	0.96	64

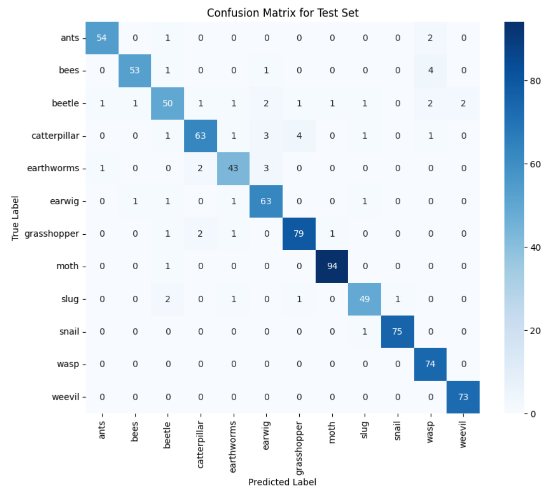
Table 2: Classification Report (Partial)

4.4 Visual Results

--- Classification Report ---

	precision	recall	f1-score	support
ants	0.95	0.96	0.96	57
bees	0.96	0.90	0.93	59
beetle	0.92	0.76	0.83	63
catterpillar	0.94	0.85	0.89	74
earthworms	0.90	0.90	0.90	49
earwig	0.79	0.94	0.86	67
grasshopper	0.95	0.92	0.93	84
moth	0.99	0.98	0.98	95
slug	0.88	0.96	0.92	54
snail	1.00	0.99	0.99	76
wasp	0.91	0.99	0.95	74
weevil	0.96	0.99	0.97	73

(a) Classification Report



(b) Confusion Matrix

Figure 2: Training and Validation Performance

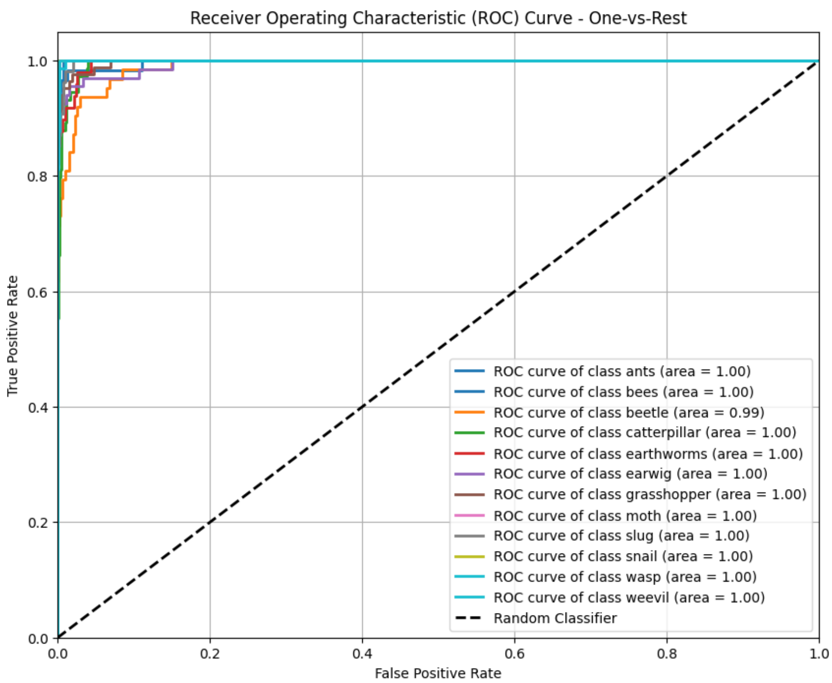


Figure 3: ROC Curves for All Classes

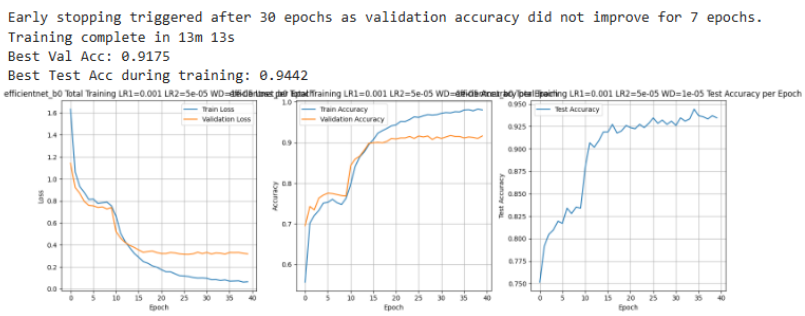


Figure 4: Acuracy loss plots

5 Outcomes

- High-performance pest classification models with:
  - Test accuracy exceeding 94%
  - Reduced overfitting and improved generalization
- A comparative study of ResNet vs. EfficientNet performance
- Insights into the impact of deeper models and dynamic hyperparameter tuning
- A reusable pipeline for agricultural image classification tasks

6 Deployment and System Demonstration

The model was deployed as a web application using PythonAnywhere and Vercel (Flask & Next.js). Users can upload images, and the model returns the predicted pest class with a confidence score. The best model was exported to ONNX format for deployment. All model files, documentation, and instructions are included in the project repository along with a comprehensive README.

Prototype Features

- User-friendly interface for farmers and agronomists
- Image upload functionality
- Display of top-3 predicted classes with confidence scores
- Accessible on mobile and desktop

Deployment Interface Screenshot

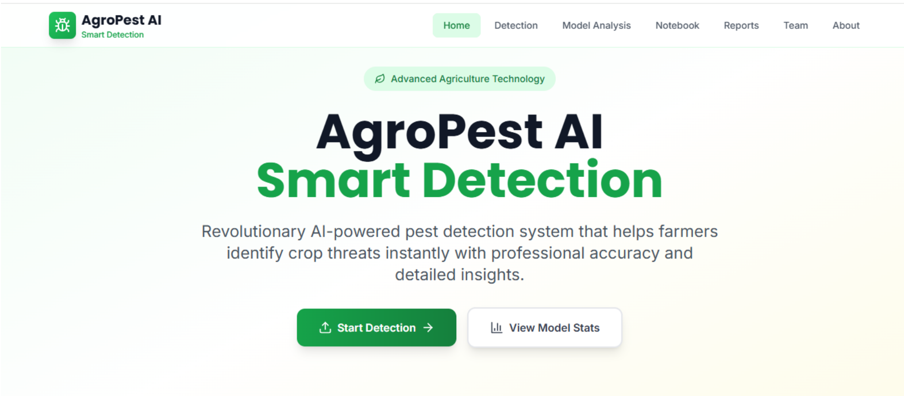


Figure 5: Web Interface for Pest Classification

Future Work:

- Expand dataset with more real-world examples
- Model quantization for edge deployment
- Active learning to retrain on incorrectly classified samples

7 Conclusion

We presented a robust computer vision system for crop pest detection using deep learning. With over 94% test accuracy and a functional deployment prototype, our system demonstrates real-world potential in aiding precision agriculture.

8 References

1. Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *International Conference on Machine Learning (ICML).*
2. Kaggle Dataset: Agricultural Pests Image Dataset – `https://www.kaggle.com/datasets/vencerlanz09/agricultural-pests-image-dataset`
3. PyTorch Documentation – `https://pytorch.org/docs/`
4. Flask Framework – `https://flask.palletsprojects.com/`
5. Vercel Deployment Platform – `https://vercel.com/`
6. PythonAnywhere – `https://www.pythonanywhere.com/`
7. Next.js Framework – `https://nextjs.org/`