PROJECT ABSTRACT

Insect Pest Identification using CNN

A transfer-learning CNN that identifies crop insect pests from field photos, trained on the IP102 dataset (Wu et al. 2019).

01. Title

Insect Pest Identification using CNN

02. Introduction / Background

Crop losses to insect pests are reduced when pests are identified early, but field identification still depends on scarce agronomist expertise. Convolutional neural networks trained by transfer learning can turn a field photo into a ranked shortlist of pest identities, making identification support widely available. The IP102 dataset (Wu et al. 2019) — a large-scale academic benchmark of crop pest photographs organized in a hierarchical taxonomy from superclass to species level — is the standard reference for this task and provides the training data. A Flask demo application serves the trained model so field photos can be identified interactively.

03. Problem Statement

Farmers and crop scouts cannot always identify insect pests quickly enough to act, and expert help is not always at hand. An identification aid is needed that takes a field photo and returns the most likely pest identities with confidence scores, trained and evaluated transparently on a public academic benchmark.

04. Objectives

Build a transfer-learning CNN that classifies crop insect pests from field photos; wrap it in a Flask demo with top-3 predictions and confidence display; log validation accuracy and loss curves during training and compute a per-class confusion matrix on the validation split, with a design target of approximately 80%+ validation accuracy; and deliver a complete viva kit (report, PPT, Q&A) alongside the code and trained weights.

05. Proposed System / Working

Pest photos from the IP102 dataset are loaded and split into stratified train and validation sets. Training images are augmented with flips, rotations, crops and color jitter to simulate field-photo conditions, while validation images are center-cropped and normalized. An ImageNet pre-trained backbone is trained in two phases: first the pest classifier head alone, then full-network fine-tuning, with validation accuracy and loss logged each epoch. The best checkpoint is exported as a .pth file, loaded by the Flask app at startup, and each uploaded field photo is preprocessed and classified in a single forward pass. Softmax scores are ranked and the top-3 pests are displayed with confidence bars.

06. Technologies / Components Used

Python 3.10 and PyTorch for transfer-learning training, validation and inference; torchvision, PIL and OpenCV for image loading, augmentation and preprocessing; NumPy and scikit-learn for metrics and the confusion matrix; Matplotlib and Seaborn for accuracy and loss curves; a Flask web app with an image-upload UI; the IP102 dataset as the training data; and trained weights exported as .pth from the included training run.

07. Key Features

Transfer-learning CNN classifier over IP102 crop pest categories; Flask demo with field-photo upload and top-3 pest predictions with confidence scores; field-condition augmentation (flips, rotations, crops, brightness jitter); training notebook logging validation accuracy, loss curves and the confusion matrix; per-class evaluation with confusion matrix computed during the build; exportable .pth weights; and configurable confidence threshold and top-k display.

08. Applications / Use Cases

Advisory support for farmers and crop scouts identifying field pests; agricultural extension and training demonstrations; pest-survey photo triage; and as a teaching example of transfer learning on an imbalanced, real-world image benchmark.

09. Results & Scope

Evaluation is designed as a transparent, buyer-run procedure: the training notebook computes validation accuracy every epoch, logs accuracy and loss curves, and builds a per-class confusion matrix on the stratified validation split. The design target is approximately 80%+ validation accuracy, computed by the buyer's own training run. Scope is limited to identification support: field-photo variability (blur, lighting, camouflage) and IP102 class imbalance are the documented weak points, and the system is an educational agri-advisory aid whose identifications must be confirmed by an agronomist before any treatment decision.

10. Conclusion

The project delivers an end-to-end insect pest identification pipeline — dataset loading, transfer-learning training, transparent evaluation and an interactive Flask demo — with all metrics computed during the buyer's own build. It demonstrates the core deep-learning workflow for agricultural image classification while remaining honest about its limits as an educational advisory aid.

11. References

Wu, X., Zhan, C., Lai, Y.-K., Cheng, M.-M., Yang, J., 'IP102: A Large-Scale Benchmark Dataset for Insect Pest Recognition', CVPR 2019; He, K., Zhang, X., Ren, S., Sun, J., 'Deep Residual Learning for Image Recognition', CVPR 2016; Deng, J. et al., 'ImageNet: A Large-Scale Hierarchical Image Database', CVPR 2009.