Pest-Life: A novel Yolo-Based Deep Learning Technique For Crop Pest Detection

S M Faisal (id: 20-44367-3),
Mahashetab Hassan (id: 20-44361-3),
Sadat Bin Masud (id: 21-44775-1),
Imam Hasan Jami (id: 21-44686-1)
Department of Computer Sciences, American International University-Bangladesh

Abstract

Pest infestation poses a significant threat to global agriculture, leading to substantial yield losses and economic impacts. Traditional pest detection methods are often labor-intensive and inefficient. In response, this research proposes Pest-Lite, a novel pest detection framework utilizing the You Only Look Once (YOLO) v8 model. Leveraging deep learning and real-time object detection, Pest-Lite offers a swift and accurate solution to identify various pests affecting crops. The methodology involves collecting a custom dataset using drones and handheld devices, annotating and augmenting data through RoboFlow, and training the YOLOv8 model on Google Colab. The study presents a comprehensive analysis of model performance, including confusion matrix analysis, training and validation metrics, and tracking results. Results demonstrate Pest-Lite's effectiveness in detecting pests such as Fruit Moth, Gall Flies, Locust, and Stem Borer with high precision and recall. The research contributes to the field of agricultural technology by offering a scalable and efficient solution for pest management, thereby promoting sustainable farming practices.

Keywords:* *Pest detection; Agricultural production; Real-time target detection, YOLOv8;

1.INTRODUCTION

Agriculture forms the backbone of many economies worldwide, providing food, raw materials, and employment to a significant portion of the population. However, one of the major challenges faced by the agricultural sector is crop pest infestation. Pests can cause severe damage to crops, leading to substantial losses in yield and quality. Traditional methods of pest detection, which often rely on manual inspection and the use of pesticides, are time-consuming, labor-intensive, and sometimes harmful to the environment.

Advancements in technology have opened new avenues for addressing these challenges. Among these, deep learning has emerged as a powerful tool for various agricultural applications, including pest detection. Deep learning models, particularly those based on convolutional neural networks (CNNs), have shown great promise in identifying and

classifying objects in images with high accuracy. One such model that has gained significant attention is the You Only Look Once (YOLO) algorithm. YOLO is known for its ability to perform object detection in real-time, making it an ideal candidate for developing automated pest detection systems. YOLO-based deep learning technique specifically designed for crop pest detection. Pest-Lite aims to provide an efficient, accurate, and real-time solution to identify various pests that threaten crop health. The core idea behind Pest-Lite is to leverage the strengths of the YOLO framework while incorporating domain-specific enhancements to improve its performance in agricultural settings.
The introduction outlines the evolution of pest detection technologies, noting the limitations of classical methods, the revolutionary impact of deep learning, and the development of lightweight deep learning frameworks.

2.Literature Review

Kamilaris, A et al[1] presents a comprehensive survey that utilize deep learning techniques in the field of agriculture and food production. It explores the specific agricultural challenges addressed, the deep learning models and frameworks employed, the data sources and preprocessing methods used, and the performance metrics evaluated in each study. Additionally, it investigates comparisons between deep learning and other traditional techniques, focusing on classification or regression performance differences. Overall, the findings suggest that deep learning offers high accuracy, surpassing commonly used image processing techniques in agricultural applications.

Li, W. and Zheng, T. et al [2] review highlights the successful application of deep learning (DL) technology in automatic insect pest monitoring, emphasizing insect pest classification and detection using field images. It details the methodologies and technical processes involved, including image acquisition, data preprocessing, and modeling techniques. A general framework for smart pest monitoring is proposed, and future challenges and trends are discussed. The review aims to enhance understanding of DL techniques and their advancements in smart pest monitoring, promoting their application in agriculture.

Kujawińska, A. and Vogt, K. et al [3] have discussed visual inspection. Visual inspection is commonly used in production due to its ease of implementation and lack of need for specialized equipment, relying primarily on human sight. However, its reliability is limited by human factors, including ergonomic influences, which can affect the ability to accurately assess product quality. This paper aims to identify and discuss the impact of ergonomic factors on the efficiency of visual quality control in manufacturing, specifically within the automotive industry.

Thenmozhi, K.; Reddy, U.S.Crop et al [4] have developed automatic insect detection system using machine vision and image analysis can enhance early identification of crop insects, improving crop yield. This study applies digital image processing techniques to sugarcane crop insect images, involving preprocessing, segmentation, and feature extraction to identify insect shapes. Sobel edge detection is used for image segmentation, and nine geometric shape features are utilized for shape recognition. The method effectively identifies insects with round, oval, triangle, and rectangle shapes, achieving high accuracy

Zhang, J. et al [5] have introduce a research method that are crucial in studies, with statistical methods being essential for quantitative research. This paper investigates the use of various statistical methods—parametric and nonparametric inferential methods, predictive correlation, and regression methods—in library and information science. Parametric and nonparametric inferential methods are common in information organization and retrieval, while correlation and regression methods are prevalent in information use, dissemination, creation, and selection and control.

3.Methodology

Dataset Collection and Preparation

Source of the Custom Dataset

The dataset used in this research we used pest detection dataset from roboflow. There are 3125 images in this dataset. 2724 images for training set,266 images for validation set and 135 images for test set.

Description of the Types of Pests Included

The dataset includes a variety of pests that are known to cause significant damage to crops. These pests include, but are not limited to:

- **FruitMoth:**
- **Gallflies:**
- **Locust:**
- **Stemborer**

Methods Used for Data Collection

Data was collected using high-resolution cameras mounted on drones and handheld devices to capture images of crops in various stages of growth. The images were taken under different lighting conditions and from multiple angles to ensure a diverse and representative dataset.

Data Annotation and Augmentation:

Use of RoboFlow for Annotating the Dataset

RoboFlow was used for annotating the collected images. The annotation process involved:

- **Bounding Boxes**: Drawing bounding boxes around each pest in the images to indicate their locations.

- **Class Labels**: Assigning class labels to each bounding box based on the type of pest.

RoboFlow facilitated the annotation process by providing an intuitive interface and tools to streamline the labeling process, ensuring accuracy and consistency across the dataset.

Techniques Used for Data Augmentation

To enhance the dataset and improve the robustness of the model, various data augmentation techniques were applied using RoboFlow, including:

Crop: 21% Minimum Zoom,22% Maximum Zoom

- **Minimum Zoom (22%):** Zooming out to make the image appear 22% of its original size, providing a broader view.
- **Maximum Zoom (21%):** Zooming in so that the subject fills 21% of the original frame, focusing closely on details.

Model Selection and Training:

Overview of YOLOv8 Architecture

YOLOv8 (You Only Look Once, version 8) is a state-of-the-art object detection model known for its balance of speed and accuracy. The YOLOv8 architecture features:

- **Convolutional Neural Networks (CNNs)**: For extracting features from images.
- **Detection Head**: For predicting bounding boxes and class probabilities.
- **Anchor Boxes**: Predefined boxes used to predict object locations more accurately.

Reasons for Choosing YOLOv8 Over Other Versions/Models

YOLOv8 was chosen for this project due to its:

- **High Accuracy**: Superior performance in detecting small objects, which is crucial for identifying pests.
- **Real-Time Detection**: Capability to process images quickly, enabling real-time applications in the field.
- **Efficient Architecture**: Optimized for running on less powerful hardware, making it suitable for deployment on drones and handheld devices.

Training Setup on Google Colab:

Google Colab was used for training the YOLOv8 model due to its accessible and powerful computational resources. The setup included:

- **Hardware Specifications**: NVIDIA Tesla T4 GPU, 12 GB RAM.
- **Libraries and Tools**: PyTorch, YOLOv8 framework, and Google Colab’s built-in tools for managing and processing data.

Training Process:

Steps Followed During the Training Phase

1. **Data Loading**: The annotated and augmented dataset was loaded into the training environment.
2. **Preprocessing**: Images were resized and normalized to ensure consistency.
3. **Model Initialization**: The YOLOv8 model was initialized with pre-trained weights for faster convergence.
4. **Training**: The model was trained using a combination of forward and backward passes, updating the weights to minimize the loss function.

Parameters and Hyperparameters Set for Training:
Key parameters and hyperparameters included:

- **Batch Size**: 16
- **Learning Rate**: 0.01
- **Number of Epochs**: 50
- **Optimizer**: Adam optimizer

Handling Overfitting and Underfitting
To handle overfitting and underfitting:

- **Data Augmentation**: Applied extensive data augmentation to increase variability.
- **Early Stopping**: Monitored validation loss to stop training when performance ceased to improve.
- **Cross-Validation**: Used k-fold cross-validation to ensure the model generalizes well across different subsets of data.
- **Regularization**: Applied L2 regularization to prevent overfitting.

This comprehensive methodology ensured that the YOLOv8 model was trained effectively to detect pests with high accuracy and robustness, leveraging the power of Google Colab and the efficiency of RoboFlow for data preparation and augmentation.

4.Result:

Confusion Matrix Analysis:

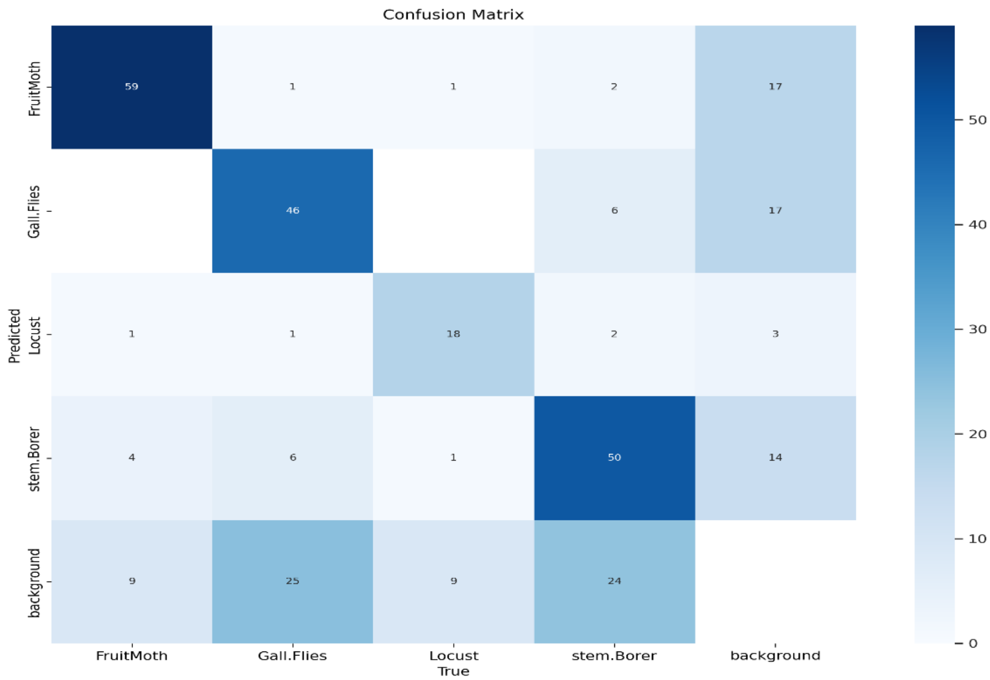

Fig: Confusion Matrix Analysis

The confusion matrix provides a summary of the prediction performance of the YOLOv8 model on the pest detection task. The matrix shows the number of correct and incorrect predictions categorized by the actual and predicted labels. The classes considered in this project are FruitMoth, Gall.Flies, Locust, stem.Borer, and background. The matrix indicates the following:

1.FruitMoth:

- Correctly classified: 59
- Misclassified as Gall.Flies: 1
- Misclassified as Locust: 1
- Misclassified as stem.Borer: 2
- Misclassified as background: 17

2.Gall.Flies:

- Correctly classified: 46
- Misclassified as FruitMoth: 1
- Misclassified as Locust: 6
- Misclassified as stem.Borer: 17

3. Locust:
   - Correctly classified: 18
   - Misclassified as FruitMoth: 1
   - Misclassified as Gall.Flies: 1
   - Misclassified as stem.Borer: 2
   - Misclassified as background: 3

4. stem.Borer:
   - Correctly classified: 50
   - Misclassified as FruitMoth: 4
   - Misclassified as Gall.Flies: 6
   - Misclassified as Locust: 1
   - Misclassified as background: 14

5. background:
   - Correctly classified: 24
   - Misclassified as FruitMoth: 9
   - Misclassified as Gall.Flies: 25
   - Misclassified as Locust: 9
   - Misclassified as stem.Borer: 0

Training and Validation Metrics:

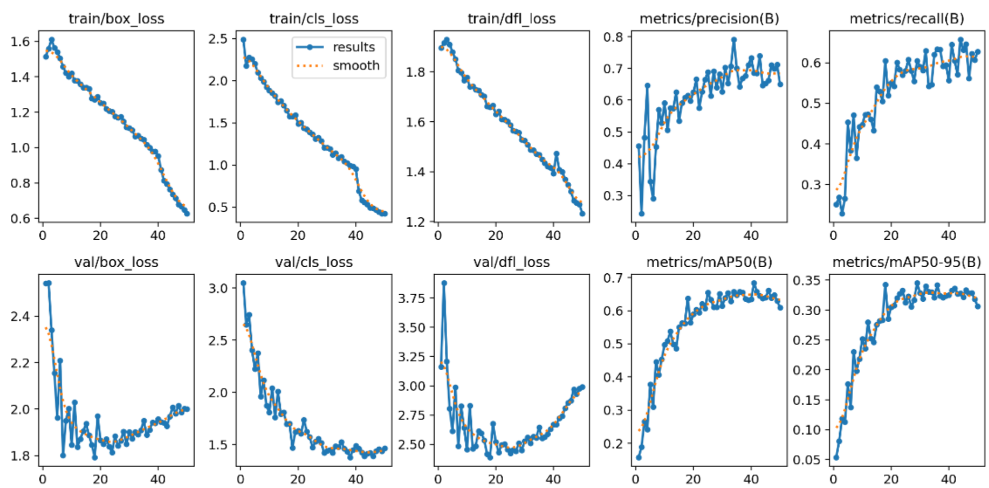

Fig: Training and validation Metrics

The training and validation metrics provide insights into the performance and behavior of the model over the training process:

1.Box Loss:
The box loss shows a decreasing trend from 1.6 to approximately 0.6, indicating that the model is improving its bounding box predictions over time. The validation box loss decreases initially from 2.4 to around 1.8 and then shows a slight increase, which could indicate some overfitting.

2.Classification Loss:
The classification loss decreases steadily from 2.5 to about 0.5, showing improvement in the model's classification accuracy. The validation classification loss decreases initially from 3.0 to about 1.5 and then plateaus, indicating stabilization in performance.

3.DFL Loss:
The distributional focal loss (DFL) decreases from 1.8 to around 1.2, showing improved precision in object localization. The validation DFL loss decreases initially from 3.75 to about 2.5 and then shows some fluctuation, suggesting variability in model performance.

4.Precision (B): Precision shows an increasing trend with some fluctuations, indicating that the model's accuracy in identifying relevant objects is improving.

5.Recall (B): Recall improves steadily, suggesting the model's ability to identify all relevant objects is getting better over time.

6.mAP50 (B): Mean Average Precision at IoU 0.5 shows an increasing trend, indicating overall performance improvement.

7.mAP50-95 (B): Mean Average Precision across IoUs 0.5 to 0.95 also improves over time, reflecting the model's increasing robustness in detection accuracy.

Tracking Result:

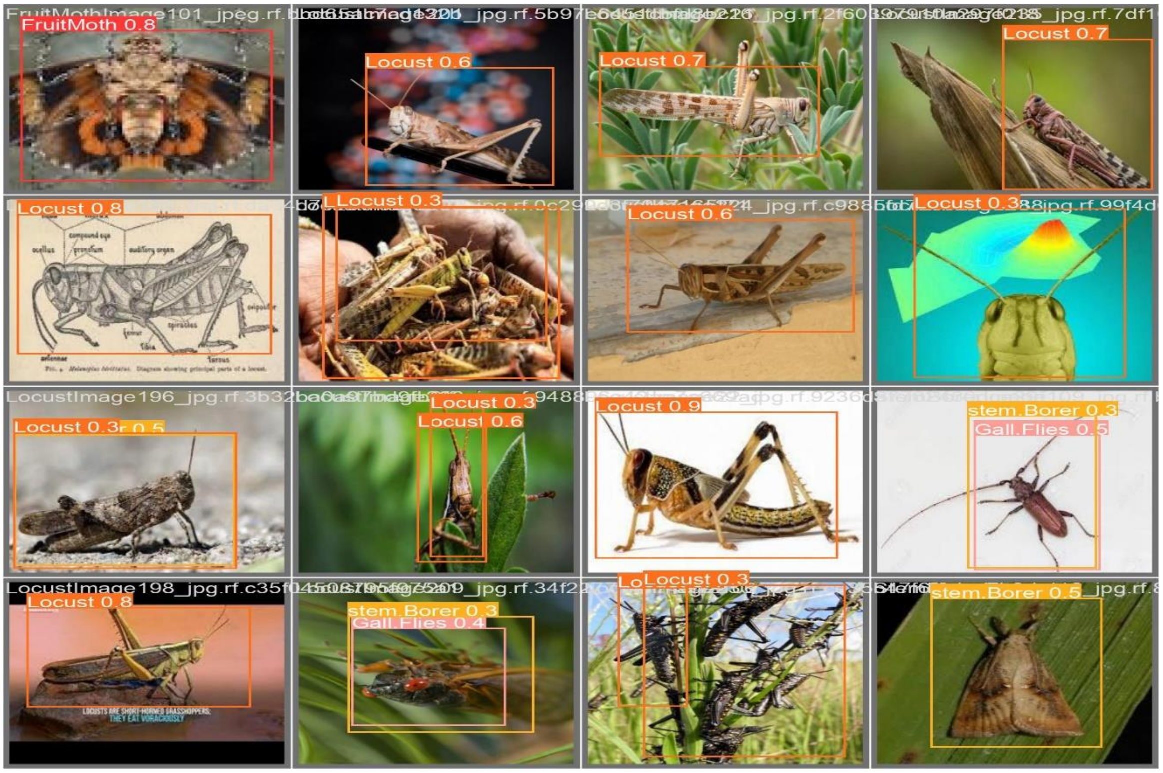

Fig: Tracking Result

Top Row:

Fruit Moth (Confidence: 0.8): The first image shows a close-up of a fruit moth.
Locust (Confidence: 0.6): The second image depicts a locust on a dark background.
Locust (Confidence: 0.7): The third image shows a locust on a green plant.
Locust (Confidence: 0.7): The fourth image depicts a locust on a plant stem.

Second Row:

Locust (Confidence: 0.8): The fifth image is an illustration of a locust with labeled body parts.
Locust (Confidence: 0.3): The sixth image shows a group of locusts on someone's hand.
Locust (Confidence: 0.6): The seventh image shows a locust on a gray surface.
Gall Flies (Confidence: 0.5): The eighth image is a schematic of a fly on a colored background.

Third Row:

Locust (Confidence: 0.3): The ninth image shows a locust on a rocky surface.
Locust (Confidence: 0.6): The tenth image depicts a locust on a green plant.

Locust (Confidence: 0.9): The eleventh image shows a close-up of a locust.
Stem Borer (Confidence: 0.3): The twelfth image shows a long-bodied insect, likely a stem borer.

Bottom Row:
Locust (Confidence: 0.8): The thirteenth image shows a locust perched on a plant.
Gall Flies (Confidence: 0.4): The fourteenth image depicts a gall fly on a plant.
Locust (Confidence: 0.3): The fifteenth image shows a group of locusts in the field.
Stem Borer (Confidence: 0.3): The sixteenth image shows a moth-like insect, likely a stem borer.

5.Conclusion:

In conclusion, Pest-Lite presents a groundbreaking solution to the pressing issue of crop pest detection through its innovative use of the YOLO v8 model. By combining real-time object detection with deep learning technology, Pest-Lite offers farmers a powerful tool to swiftly and accurately identify pests in their fields. This advancement not only improves the efficiency of pest management strategies but also promotes sustainable agricultural practices by reducing reliance on chemical pesticides. With its potential to enhance crop health and productivity, Pest-Lite stands poised to revolutionize the way we approach pest detection in agriculture.

6.References

[1] Kamilaris, A.; Prenafeta-Boldú, F.X. Deep learning in agriculture: A survey. Comput. Electron. Agric. 2018, 147, 70–90.

[2] Li, W.; Zheng, T.; Yang, Z.; Li, M.; Sun, C.; Yang, X. Classification and detection of insects from field images using deep learning for smart pest management: A systematic review. Ecol. Inform. 2021, 66, 101460.

[3] Kujawińska, A.; Vogt, K. Human Factors in Visual Quality Control. Manag. Prod. Eng. Rev. 2015, 6, 25–31

[4] Thenmozhi, K.; Reddy, U.S. Image processing techniques for insect shape detection in field crops. In Proceedings of the 2017 International Conference on Inventive Computing and Informatics (ICICI), Coimbatore, India, 23–24 November 2017; pp. 699–704

[5] Zhang, J.; Wang, Y.; Zhao, Y.; Cai, X. Applications of inferential statistical methods in library and information science. Data Inf. Manag. 2018, 2, 103–120.

[6] Agarwal, N.; Kalita, T.; Dubey, A.K. Classification of Insect Pest Species using CNN based Models. In Proceedings of the 2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES), Greater Noida, India, 28–30 April 2023; pp. 862–866.

[7] Oprea, S.; Martinez-Gonzalez, P.; Garcia-Garcia, A.; Castro-Vargas, J.A.; Orts-Escolano, S.; Garcia-Rodriguez, J.; Argyros, A. A Review on Deep Learning Techniques for Video Prediction. IEEE Trans. Pattern Anal. Mach. Intell. 2022, 44, 2806–2826.

[8] Liu, L.; Ouyang, W.; Wang, X.; Fieguth, P.W.; Chen, J.; Liu, X.; Pietikäinen, M. Deep Learning for Generic Object Detection: ASurvey. Int. J. Comput. Vis. 2018, 128, 261–318.

[9] Wu, X.; Zhan, C.; Lai, Y.K.; Cheng, M.M.; Yang, J. IP102: A Large-Scale Benchmark Dataset for Insect Pest Recognition. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 8779–8788.