Machine Learning Based Prediction of hexapod invertebrates and Its Impact on Biodiversity

by

Anurag Sikder
24341164
Opshora Noshin Eshika
21301312
Iffat Haque Mithila
21301143
Shahed Abdullah
21301128

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
October 2025

Declaration

It is hereby declared that

1. The thesis submitted is my/our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Anurag Sikder	Opshora Noshin Eshika
24341164	21301312
Iffat Haque Mithila	Shahed Abdullah
21301143	21301128

Approval

The thesis titled "Machine Learning Based Prediction of Hexapod Invertebrates and Its Impact on Biodiversity" submitted by

1. Anurag Sikder(24341164)
2. Opshora Noshin Eshika(21301312)
3. Iffat Haque Mithila(21301143)
4. Shahed Abdullah(21301128)

Of Summer, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science and Engineering in October 10, 2025.

Supervisor:
(Member)

Mr. Md. Tawhid Anwar

Senior Lecturer
Department of Computer Science and Engineering
Brac University

Co-Supervisor:
(Member)

Mr. Md. Sabbir Ahmed

Lecturer
Department of Computer Science and Engineering
Brac University

Thesis Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD

Chairperson and Associate Professor
Department of Computer Science and Engineering
Brac University

Abstract

Insects represent one of the most diversified groups and are also critical to maintaining ecological balance; however, their monitoring is limited by traditional field sampling and manual identification techniques. This research introduces a new framework that combines automated insect detection with biodiversity quantification and hotspot visualization, converting the recognition results into ecological conservation decision-support tools. We have used a dataset of 6,000 annotated images of ten equivalent classes of hexapods to train and evaluate the state-of-the-art systems, YOLOv8 and YOLOv10, in a hybrid setup. The YOLOv8 model showed better results in all the tested detection metrics. Biodiversity was also determined by ecological metrics besides detection accuracy, which also showed that the site had a species richness of 10, a Shannon diversity index of 2.296 and an evenness value of 0.997. To find diversity gradients across locations, numerical indices were replaced with a hotspot analysis. The maps indicate sites that need to be preserved right away, not places that can be restored environmentally. The study shows that combining advanced deep learning methods with biodiversity measurements can lead to data-rich, scalable, and ecologically sound insights for monitoring and making policy decisions.

Keywords: YOLOv8, YOLOv10, Shannon Index, Evenness, Hotspot

Acknowledgement

Firstly, we would like to express our heartfelt gratitude to the Almighty, whose boundless mercy, guidance, and blessings have accompanied us throughout this journey. Without His divine grace, this achievement would not have been possible. We would like to extend our sincere appreciation to our respected supervisor, Md. Tawhid Anwar, for his guidance, valuable suggestions, and support during the course of our research. Our deepest gratitude goes to our co-supervisor, Md. Sabbir Ahmed, whose continuous guidance, hands-on support, and insightful feedback have played a pivotal role in the successful completion of this thesis. His patience, encouragement, and dedicated involvement throughout every stage of our work have been truly invaluable.Finally, we would like to thank all faculty members, friends, and well wishers who have directly or indirectly supported us in completing this research successfully.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Hexapods are the most widespread group of creatures on Earth [15]. There are over one million described species of hexapods, with many more undescribed species [15], so hexapods are found in almost every terrestrial and freshwater environment [9]. They play a vital role in the ecology by performing critical functions like pollination, decomposition, cycling of nutrients, and food web regulation. Therefore, hexapod diversity is the topic of research that is of utmost importance in ecological research, conservation policy, and sustainability efforts. The traditional approaches of hexapod surveillance are specimen collection, manual identification, and field surveys [14]. Despite being time-consuming and labor-intensive, these methods are not easily scalable; however, they are very taxonomically accurate [8]. These methods are prone to the oversampling of larger or morphologically defined taxa and the undersampling of small or cryptic species [3]. As the number of insects has been decreasing nowadays and the percentage of hexapods exposed to loss of habitats, pollution, and climate change grows [19], a wider-ranging monitoring plan is highly desired. Artificial intelligences like computer vision and deep learning have potent instruments to automate the monitoring of biodiversity [17]. YOLO models are very successful in detecting small and morphologically diverse organisms [23]. Even though these methods are starting to find their application in ecological research, the majority of the available uses are limited to basic classification processes. There are limited systems that combine automated hexapod detection with indicators of ecology or conservation, and this is an issue of vital concern to biodiversity informatics [24].

1.2 Research Problem

The Deep learning-based hexapod monitoring is a fairly new development, but its use in ecological settings is limited. [16]. Deep learning has significantly enhanced the precision of species identification. However, the problems are still there.

Firstly, most of the currently existing systems are mainly aimed at improving the quality of raw categorization, but many of the important measures of biodiversity, including richness, evenness, and variety, are not disregarded [24]. Considering the lack of these ecological measures, the importance of the results that are obtained through automated monitoring to ecological studies is considerably reduced.

Secondly, the existing hexapod information is often biased, and there is a tendency to give more significance to large-bodied organisms[17]. This has the effect of reducing the efficiency of the models in detecting smaller or morphologically related species, leading to an undisguised representation and hence making automated detection methods ecologically ineffective.

Thirdly, to make a wise decision on conservation and restoration of biodiversity hotspots, conservation planning must obtain certain information, e.g., information about biodiversity hotspots [21]. Spatial pattern analysis is not a routine procedure in most AI-based detection approaches and observations of hexapods are not often linked to real conservation efforts [22].

To conclude, deep learning model can enhance hexapod identification, but their use does not frequently give ecologically significant results. Based on this, automated monitoring is becoming more difficult to combine with conservation planning and the development of policies that are environmentally favorable.

1.3 Research Objective

Monitoring insect diversity is vital for understanding ecosystem health, agricultural sustainability, and biodiversity conservation. However, manual identification of hexapods is time consuming, error-prone, and often restricted by taxonomic expertise. This research aims to develop an automated framework that integrates advanced deep learning-based object detection (YOLOv8 and YOLOv10) with biodiversity metrics to enable accurate, scalable, and interpretable insect monitoring.

Objectives

- To train and evaluate YOLO models on a curated dataset of 10 insect classes.
- Using of standard metrics like F1-score , Precision, Recall and most important metric topp YOLO model mAP to find best possible value.
- To compute ecological diversity indices (Species Richness, Shannon Index, and Evenness) from detection outputs.
- To interpret model predictions for transparency and interpretability. using LIME which is an explainable AI.
- To visualize biodiversity gradients and identify ecological hotspots for conservation priority.

By bridging deep learning and ecology, this work contributes toward real-time, data-driven biodiversity assessment and sustainable ecological management.

1.4 Methodology in Brief

The present study is centered on the hexapod detection using a hybrid deep learning model integrating YOLOv8 and YOLOv10, along with the analysis of species occurrence frequencies for the assessment of biodiversity.

1. The dataset was created by the researchers who, apart from collecting, also annotated 6,000 insect images (60 classes, 600 per class). Images originally sized 640×640 pixels were normalized. Data division into training and testing folders was made with the ratio being 80 and 20 percent respectively.

2. The YOLOv8 and YOLOv10 were both optimized for the AdamW optimizer, with their learning rate being set to cosine annealing, whose schedule begins at $1 \times 10{-}3$. The heads of the detectors that are independent of anchors provided high-quality localization of tiny insects.

3. The training of YOLOv8 was performed in a Kaggle environment on an NVIDIATesla T4 GPU, whereas for high-resolution experiments, YOLOv10 was trained on an RTX 3090 workstation locally.

This procedure combines AI-based detection and ecological interpretation, thus providing a bridge between the automated evaluation with the biodiversity of insects.

1.5 Motivatations and Challenges

Motivatations

- **Biodiversity Tracking:** This feature can be used in ecological research to automatically track large populations of insects.
- **Pest Management:** Agriculture industries use the system for controlled agricultural processes.
- **Conservation Planning:** Biodiversity hotspots identification is a crucial aspect that allows restoration and protection of the ecosystems.
- **Integration of Explainable AI :** Clearly promotes understandable systems and encompassing the analysis of ecology.

Challenges

- **Skewness in Dataset:** Insect class will have impact on the generalization capability of the model negatively.
- **Small Object Detection:** Due to the nature of the techniques to identify tiny objects, the visual similarity and tiny size of insects provide a source of duplication in detection.
- **Changeable environments:** When it comes to environments, a decrease in temperature, motion blur, and occlusion has a negative influence on detection accuracy.
- **Hardware Constraints:** The training of models that require large-scale computing platforms like YOLOv10 dependent on large-scale computers.

The challenges were not reduced by the use of the iterative optimization model and hybrid assessment approaches, which made the model more reasonable and interpretable.

Chapter 2

Literature Review

2.0.1 Detection using YOLOv8 & YOLOv10

The YOLO models have made a big difference in how well we can identify objects lately. These models can find things in real time with a lot of accuracy. [12] Redmon et al. were the first to use the YOLO framework, which made it possible to find objects by solving a single regression problem. In one forward pass, this new feature allowed the model to predict both bounding boxes and class probabilities. This made it faster than before while still doing well compared to older multi-stage detectors like R-CNN. YOLOv10 introduces further improvements to scaling models, methods for object detection without anchors, and integration with newer loss functions. This makes it more accurate and helps it conclude faster on large, complex datasets.

The development of YOLOv8 and YOLOv10 represents significant advancements in automated biodiversity monitoring. Its implementation is also significant in recent times. They are a great solution for ecological studies as their result are very accurate. It can work in real time and also can uses AI methods that are very understandable. This is significantly true for ecosystems that are hard to monitor. As an ecosystem have a lot of different species.

2.0.2 Foundations of Information Theory

Claude Shannon's 1948 article "A Mathematical Theory of Communication" is widely considered to be the first important work in the field of information theory. He came up with many number of important ideas. For example, **entropy, ergodicity, channel capacity** and **fidelity.** These ideas created a new scientific way to think about communication among species.

Following the rules of continuity, monotonicity, and additivity, entropy gives us a way to measure how uncertain an information source is. The Asymptotic Equipartition Property (AEP) is what makes long-term communication possible. It lets anyone encode common sequences in a way that works well. Shannon's channel capacity theory says that the fastest rate at which information can be sent with a very low error rate is the maximum. It also mentions that efficient communication can only be achieved when the entropy of the source is less than the channel capacity [1]. He also made the notion of fidelity more formalized through the ratedistortion theory, thus making it possible to specify controlled tolerances to transmit continuous signals in a noisy environment. Such contributions not only contributed towards enriching theoretical knowledge but also formed the basis of current loss compression methods and useful information funtions.

2.0.3 Quantitative Measurement of Biodiversity

In a landmark article, published in 1966 [2], E.C. Pielou offered a strict procedure for measuring the diversity of species in ecological collections. In his work, Pielou coined mathematical indices, including the Shannon Weiner index and the index of Simpson, to reveal his applicability in ecological surveys that are unequally sampled, e.g., taxonomic and quadrat-based surveys. The manuscript has developed a strict statistical context by clearly defining the terms richness and evenness. It also provides the direction for choosing the proper metrics depending on the goals of the research. Based on information theory, the Shannon-Weiner index was modified to be applied to ecology. Pielou also explains the inherent shortcomings of diversity indices in that they are sensitive to sample size, and that heterogeneous populations are difficult to deal with. [2] Even though such theoretical assumptions cannot necessarily capture the complexity of the phenomenon of spatial heterogeneity or the likelihood of observing rare species, the works by Pielou help to lay the groundwork for further biodiversity analyses.

Despite these theoretical assumptions, which may not fully capture complexities like spatial heterogeneity or the detection probability of rare species, Pielou's work laid the foundation for subsequent biodiversity assessments. It significantly influenced conservation biology and ecosystem management, emphasizing the need for context-specific diversity measures and inspiring further development of advanced statistical methods for quantifying functional and phylogenetic diversity.

2.0.4 Biodiversity Assessment and Ecosystem Functioning

Basset, Novotny, Miller, and Kitching's (1996) study on the issue of measurement of the effects of the disturbance of forest on Brazilian tropical invertebrate communities is a groundbreaking one considering the issues that it tackles [3] . Quantification of biodiversity responses, the authors underline, is complicated by the fact that the extent of ecological heterogeneity is extensive and that by the time of invertebrates, the richness of the species is remarkable. They describe protocols for standard sampling aimed at increasing study comparability and explain the effects of anthropogenic stressors, including logging and habitat fragmentation, on invertebrate assemblages. The use of diversity metrics is also discussed in their analysis with references to such indices of Pielou (1966) [2], and the importance of long-term monitoring that is required to identify changes over time. Although the paper addresses the sensitivity of invertebrates in monitoring ecological systems, knowledge gaps based on functionalities and interactions between species that are important in conservation planning are also noted. Generally, this study adds to an effective evaluation of biodiversity in disturbed tropical ecosystems.

In a long-term experiment on a grassland ecosystem in a Cedar Creek Ecosystem Science Reserve, Tilman et al. (2001) [4] examined the diffrences between species diversity and ecosystem productivity. Through the manipulation richness of plant species in the experimental plots, the study proves the positive association between diversity, biomass production and ecosystem stability. According to the authors, these effects can be explained by the presence of niche complementarity, when different species can effectively exploit resources, which leads to a decrease in competition and an increase in the activity of the ecosystem. It builds on the work done on the measures of diversity by Pielou (1966), studies by Basset et al. (1996) on the disruption response of invertebrates, respectively, not to deny biodiversity as an important functional aspect of an ecosystem in terms of ecological services like productivity and resilience. Even though the results can be extrapolated to grasslands alone, the results of the study are very useful as far as conservation actions and general management of the ecosystems are concerned and additional research on multitrophic interaction, especially the effects it has on either herbivores or on microorganisms formed in soils, is required.

2.0.5 Biodiversity Monitoring and Computer Vision Benchmarks

A thorough examination of ecological variety and biodiversity surveillance was carried out by Spellerberg and Fedor [5], who brought together theory and practice in their work. They examine how the indices of diversity of Pielou (1966) [2] enable us to characterize quantitatively the notion of diversity, evenness and the composition of communities among the classes of ecosystems. They underline that in trying to have uniform monitoring, it is vital to monitor changes in biodiversity, especially where there are pressures like loss of biodiversity on a habitat, or such as due to climate change. The statistical techniques and sampling protocols that can be adopted for a wide new species of plants and invertebrates were not to be found in the book. Rather, it is the graphical representation of the research conducted by Basset et al. (1996) [3].made. It also emphasizes the ecological importance in real life, which parallels findings of Tilman et al. (2001) [4]. The book demonstrates the ability of tracking programmes to provide conservation policy with useful data by relying on real cases studies, even though it is difficult to generalize local research to global trends, solve the unknown taxa, and identify long-term trends.

Magurran (2004) [6] builds upon previous literature to provide a structured and updated approach to measuring biological diversity. The book focuses on conceptual and methodological issues, including species richness, evenness, and species abundance modeling. Magurran critically reviews older diversity measures, such as Shannon and Simpson indices, and highlights their limitations in cases of skewed abundances or non-uniform sampling. She also discusses the comparison of communities across space and time, including β-diversity, similarity/dissimilarity measures, and temporal changes, which are vital for assessing environmental change and extinction risks.

2.0.6 Advances in Object Detection, Interpretability, and Hexapod Biodiversity

Leather's (2018) [14] findings reinforce the value of standardized biodiversity monitoring and suggest potential applications of predictive models such as LIME to improve understanding of species distribution patterns in dynamic habitats. A rather comprehensive estimation of arthropod variety has been proposed by Stork (2018) [15]. He considers research that has quantified biodiversity with time, with improvements on earlier amounts presented by Pielou (1966) [2], and points out the loopholes in tax and sampling selection. It is the value of arthropods to ecosystems that causes the paper to emphasize the importance of increased focus on their monitoring and the fact that contemporary ways to measure biodiversity, including LIME, may enhance the scale of biodiversity measurements across the world.

2.0.7 Deep Learning and Explainable AI for Insect Detection

Ribeiro et al. [13] (2016) introduced **LIME (Local Interpretable Model-Agnostic Explanations)** to address the challenge of understanding black-box machine learning models. LIME works by perturbing the input data locally and fitting an interpretable surrogate model, such as a linear regression, to approximate the behavior of the complex model in that local neighborhood. The approach eases the identification of characteristics that are of the most significance to the projection of a person. The case of LIME has been demonstrated in text and image classifier research that has demonstrated its ability to provide intelligible results that can be understood by human beings without having to make changes to the underlying model. Of particular importance, the authors emphasized that LIME became extremely general and could be applied to any classification scheme, which is a significant strength of the tool. Through LIME, researchers can understand the argument that a deep learning system, including a convolutional neural network (CNN) or a YOLO based detector, makes to classify an image as a part of a specific insect species. This is necessary in ecological surveillance because it empowers trust in AI based biodiversity evaluations. Initially, in 2023, Jocher et al. introduced the next generation object-identification framework,**YOLOv8**, by Ultralytics [26] YOLOv8 was actually designed to undertake real time detection, segmentation and classification assignments, due to its potential of providing the most accurate results in the use of fast processing power. In order to make the framework more efficient at identifying small objects, the architecture makes use of advanced data-augmenting methods alongside an anchor-free detection paradigm. Furthermore, YOLOv8 uses end-to-end PyTorch pipelines, thus allowing a natural evolution of including acceleration and CPU usage to speed training and inference processes. Presumptions have already shown that YOLOv8 is more effective than its predecessors, such as YOLOv7, in mean Average Precision (mAP) and with low inference latency. Therefore, YOLOv8 performs very well with small and camouflaged insects like rice yellow stem borers when subjected to adverse field conditions.

Li et al. (2023) [27] proposed **YOLOv10**, which is an object-detection model that is lightweight and can be trained entirely. Also, it strikes a balance between speed and accuracy and adds better feature pyramids and attention systems, which enable it to detect very small and mid-sized objects more precisely. Based on common datasets, the authors presented the fact that the YOLOv10 is capable of competitive accuracy as well as effective on devices with low resources, including mobile phones. The paper has identified the capability of YOLOv10 to be used as a means of real-time ecological surveillance, with the potential to detect insects on a farm field in real-time, whilst having limited computational resources.

Providing an overview of the progress, opportunities and issues, Weng et al. (2022) [25] offered a brief review of **deep learning to biodiversity monitoring**. The study observed the potential uses of deep learning in acoustic monitoring, video tracking, and species detection based on an image. It was also proposing some solutions such as ensemble methods, enhancement of data and transfer learning as a solution to short datasets, proliferation of classes and flexible domains. To be able to make comprehensible ecological predictions, the authors emphasized the necessity of the application of explainable AI approaches such as Grad-CAM or LIME. The article is a foundation of combining accurately and clearly the object detection problems by means of deep learning with insect classification issues. Shorten and Khoshgoftaar (2019) [20] (2019) conducted a study on the **image data augmentation for deep learning**. To minimize the impact of overfitting in small sizes of data sizes, the study compared both orthodox and state-of-the-art augmentation techniques, including rotation, flipping, scaling, Mixup, Cutmix and GAN-enhanced techniques. The authors emphasized that under the ecological model, in aspects of limited datasets, the right procedure of augmentation is crucial in augmenting the deep learning models. The augmentation in insect identification tasks is seen to enhance the ability of the improved model to be resilient to issues associated with varied field environments. This is done through realistic contrast in insect direction, size and in the background.

2.0.8 Foundational Work in Information Theory and Biodiversity Metrics

Shannon's Information Theory

The article *A Mathematical Theory of Communication* (1948) [1], by Claude Shannon in 1948 and forms the beginnings of a new stirring thing that brought a tremendous impact on the fields of mathematics, engineering, and computer science. Entropy and properties of the functions, channel capacity, and fidelity are the most prominent ideas that Shannon published and they established a new scientific vocabulary of communication theory. Entropy gives an objective structure of determining the measure of uncertainty of an information source and follows a continuity principle, monotonicity, and additivity. The Asymptotic Equipartition Property (AEP) defines the fact that those sequences that a source uses with great frequency can be encoded with efficiency, with the result that viable communication protocols can be created. A theory called the channel capacity theory, as expressed by Shannon, shows that the amount of information that can be easily transmitted at a constant rate with reduced error happens due to a reduction in the correlation between errors; the theory states that reliably transmitted information is guaranteed when the source entropy does not surpass the channel capacity. Besides, Shannon established the concept of fidelity in terms of rate distortion theory to enable tolerances to be permitted in the transmission of signals in the real world. These theoretical developments formed the basis of the theoretical ideas of contemporary lossy compression algorithms as well as the development of advanced communication systems.

Pielou's Diversity Measurement

Pielou, in a work named *The Measurement of Diversity in Various Types of Biological Collections* is one of the fundamental works in the ecological literature [2].In this essay, Pielou outlines a set of mathematical indices that measure biodiversity, namely, the **Shannon-Weiner** and **Simpson** indices, thus providing a solid statistical perspective that measures the relationship between species richness and evenness. The paper identifies the difference between richness, which can be described as the simple number of unique species, and evenness, which can be described as the higher proportion of individuals among these species, and provides an orientation on the choice of perfect measures to take depending on the aims of a particular research. Even though Pielou acknowledges the constraints of these indices, many elements wherein challenges in reporting and assemblage heterogeneity are concerned, her works have significantly influenced the ecosystem management and conservation of biodiversity

Basset et al. on Tropical Invertebrate Biodiversity

In the research carried out by Basset, Novotny, Miller, and Kitching (1996) [3], the study, entitled *Assessing the Impact of Forest Disturbance on Tropical Invertebrates: Some Comments*, emphasizes all the challenges that are faced in the quantification of biodiversity responses in a community of invertebrates in its tropical environment. The authors consider the problems that are created by ecological variability and mismatched numbers of species, and they argue that a comparison between the results achieved by two different studies should be easier on the basis of standardized sampling procedures. They explain the influences of disturbance of the invertebrate communities by logging and habitat fragmentation, and add to the previous statement that the available methods are poor, as they have a poor taxonomic resolution and use of simplified diversity indices. Exemplifying the essentiality of the indices introduced by Pielou (1966) [2], the authors emphasize the need to monitor the community structure over a long period of time so that it is capable of capturing changes in the temporal context of the community. Therefore, the study provides an effective framework to evaluate biodiversity in disturbed tropical ecosystems and provides a solid guide to the conservation management of endangered invertebrates.

Chapter 3

Methodology

3.1 Methodology Overview

Top Level Overview of the proposed system (YOLOv8)

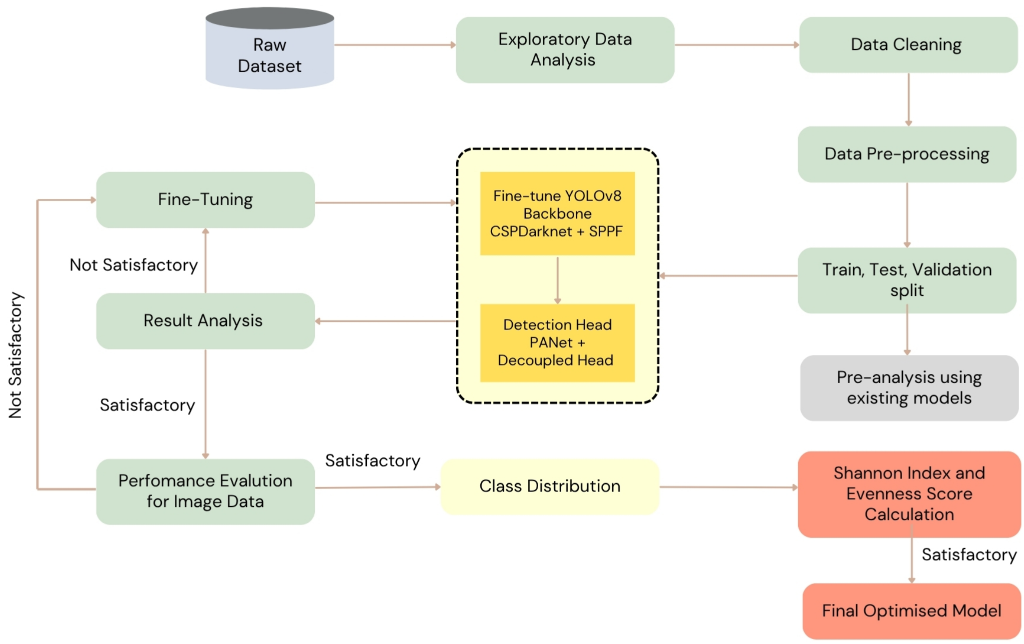

Figure 3.1: Overview of the proposed system (YOLOv8)

Top Level Overview of the proposed system (YOLOv10)

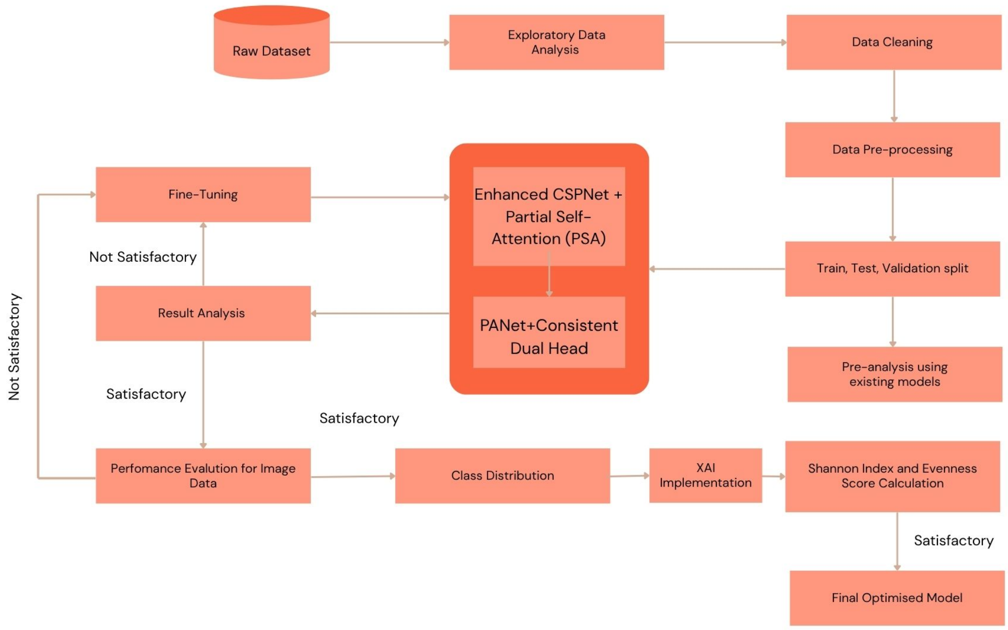

Figure 3.2: Overview of the proposed system (YOLOv10)

3.2 Architecture Design

3.2.1 Analysis of existing Architecture

Before finalizing our proposed system, we analyzed numerous deep learning architectures often utilized for ecological monitoring and object detection applications.

Rationale Behind Model Selection:

- **YOLOv8:** It is introduced by Ultralytics and is one of the latest model. YOLOv8 is good for lightweight detection and uses convolutional, C2f modules alongside anchor-free detection heads and SPPF pooling. This is why it is highly effective in detecting diminutive insects with irregular body shapes.
- **YOLOv10:** With an improved variant, YOLOv10 incorporates decoupled detection heads and enhanced small-object detection. However, it needs more computational power and is used for better accuracy.

Key Observations:YOLOv10 demonstrated improved robustness in complex environments, as it required significantly more training resources. To sum up, YOLOv8 provided the optimal balance between accuracy and speed, providing faster predictions and achieving high mean Average Precision (mAP) in all insect classes. The incorporation of LIME further substantiated the biological significance of YOLOv8's Predictions, hence it is strengthening their applicability to monitoring biodiversity.

3.2.2 Model Architecture

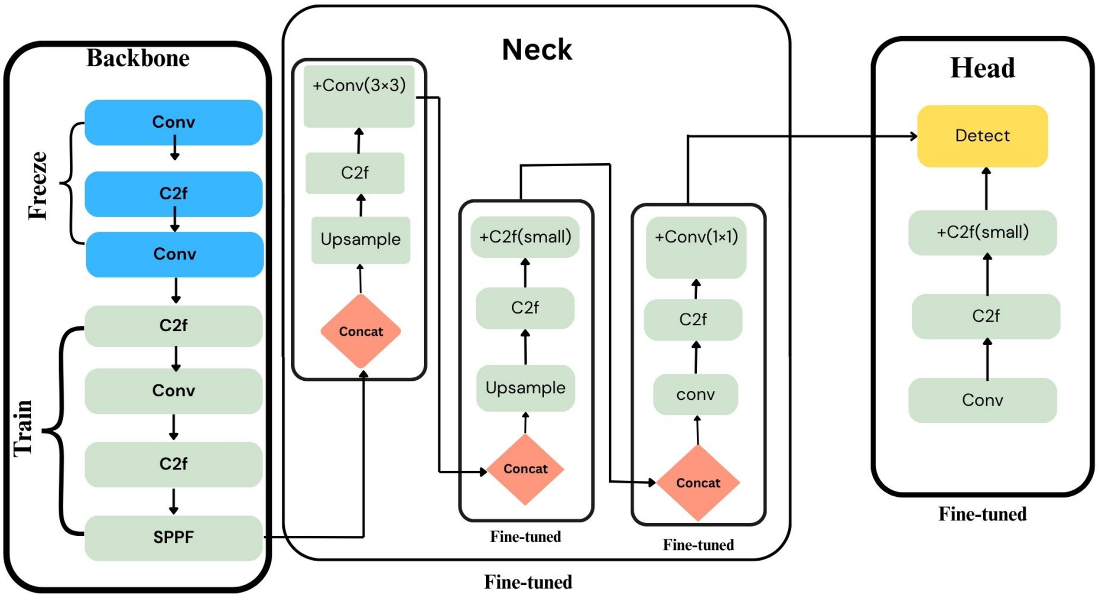

Figure 3.3: YOLOv8 Model Architecture

Representation of Input The suggested method takes an input of RGB images of insects and resizes them to a pixel of $640 \times 640 \times 3$. Here, this scaling makes a good balance between preserving detail for detecting smaller insects while maintaining computational accuracy. Moreover, the normalization and adaptive padding are applied to the input to ensure consistent feature scaling and aspect ratio preservation.

Backbone Feature Extraction The Backbone consists of multiple convolutional layers interleaved with cross stage partial connections (C2f modules). Here, two convolutional layer and one C2f module are frozen so that the weights wont be updated. So that we can extract basic low level features from images.Then, from the second C2f module, the training are happening and weights are updated by the dataset allwoing fine tuning. And this is useful for detecting objects at different sizes.

Neck: Feature Aggregation In the neck which consists 3 blocks where the output from SPPF from Backbone layer comes to first block concat. Then it goes to upsample layer, where resize lower resolution maps to match higher resolution maps. and then from C2f it goes to +conv(3x3) this is for, residual addition for better gradient flow. The output from +conv(3x3) goes to Concat of second block and then the out put of the second block goes to third blokcs concat.then +conv(1x1) the ouptu goes to detect which is from Head layer.

Head: Detection Layer The Head employs an anchor free detection mechanism that directly predicts bounding box coordinates. IT has Convolutaional part and c2f part to be fully trainable. It's Purpose is producing final predictions which are Bounding boxes, Class probabilities,Objectness scores. It Takes fused multi scale feature maps from Neck.

Loss Function and Training Objective YOLOv8 optimizes a composite loss function comprising:

- **Bounding Box Regression Loss (IoU-based, e.g., CIoU/DIoU)** to align predicted boxes with ground truth,
- **Objectness Loss** to discriminate insect presence from background regions,
- **Classification Loss (cross-entropy)** to assign species labels correctly.

This combination ensures accurate localization, confidence estimation, and class assignment.

Benefits for Ecological Applications

- Anchor-free detection improves accuracy for small and overlapping insects,
- Lightweight convolutional backbone with C2f and SPPF enables fast inference even with moderate hardware,
- Strong performance on small-object detection makes it highly effective for biodiversity monitoring.

3.3 Data Preparation

3.3.1 Data Collection

The dataset used in this research was collected directly from a horticulture officer in Bangladesh. It represents a primary and authentic source of entomological field data. The dataset was also labeled by the horticulture officer so that the dataset remains Primary source. The dataset consists of 6,000 images belonging to 10 insect classes, with approximately 600 samples per class.The labeling process was entirely manual. Using the `LabelImg` and `Roboflow` tools, each image was annotated with bounding boxes corresponding to the insect's visible body region. Class labels were assigned based on morphological features such as wing pattern, body segmentation, and antenna structure, under supervision from the horticulture officer. All annotations followed the YOLO compatible format, where each image corresponds to a `.txt` file containing class index and normalized bounding box coordinates (center-x, center-y, width, height).

The data were organized into a three level folder structure:

- **Train:** 80% of total data (4,800 images)
- **Test:** 20% of total data (1,200 images)

All images were stored in the `JPEG` format with an initial resolution of 1280×720 pixels. Metadata such as collection date, crop type and insect behavior were stored in a separate CSV file for ecological analysis. An example of the raw image and its corresponding labeled version is illustrated below.

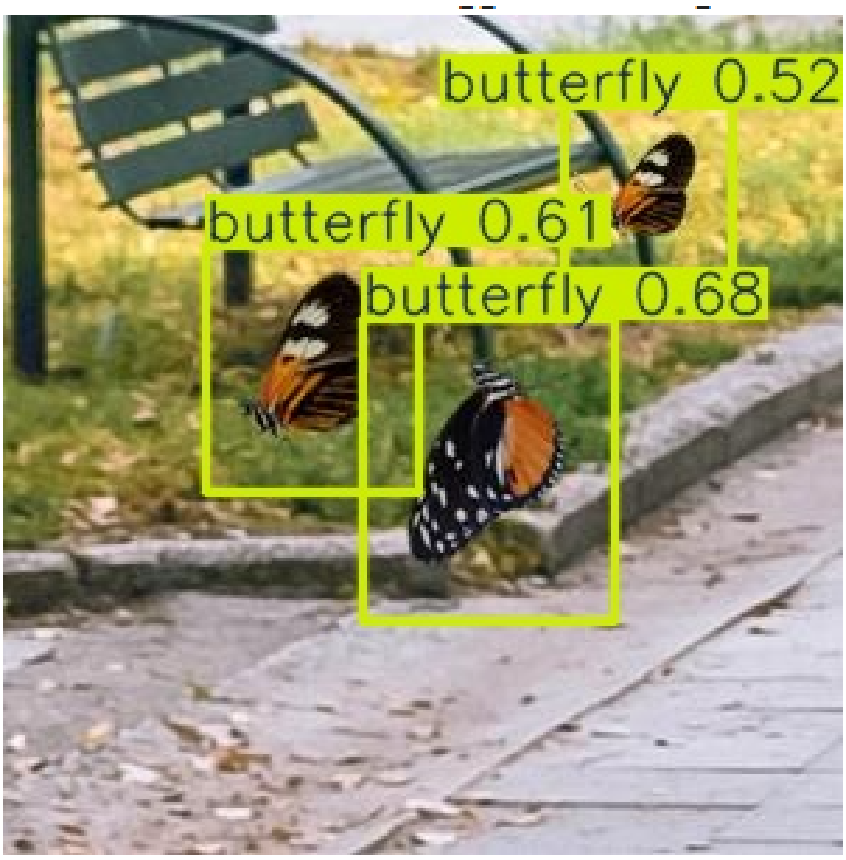

Figure 3.4: Example of insect image and its YOLO bounding box annotation.

3.3.2 Exploratory Data Analysis

We performed EDA to examine data distribution before pre-processing, the balance of classes, and visuality. This histogram revealed that most classes had representative comparison, some insects like *Rice Yellow Stem borer* and *Ants* appeared more than other insects. Brightness of image and colorization intensity were analyzed for proper normalization prior to model training. Additionally, several environmental factors were noted:

- Some samples contained motion blur due to insect movement.
- A few classes overlapped visually, This required careful labeling validation.
- Background complexity varied significantly, especially in outdoor conditions.

These observations guided the augmentation strategy to increase dataset diversity and model robustness.

3.3.3 Class Distribution

We have used a curated dataset of 6000 images of insects from 10 different species, including RiceYSB, Bee, Grasshopper, Moth, Beetle, Cicada, Dragonfly, Butterfly, Antand Cockroach. Here, these high-resolution pictures with gathers and labeled by horticulture officers of Bangladesh, hence this evaluates our authentication and makes the data primary. Every class has exactly 600 images, explaining the major requirements of object detection for machine learning. Moreover, the dataset assures the proper ecological analysis, including biodiversity hotspot conservation among all 10 classes.

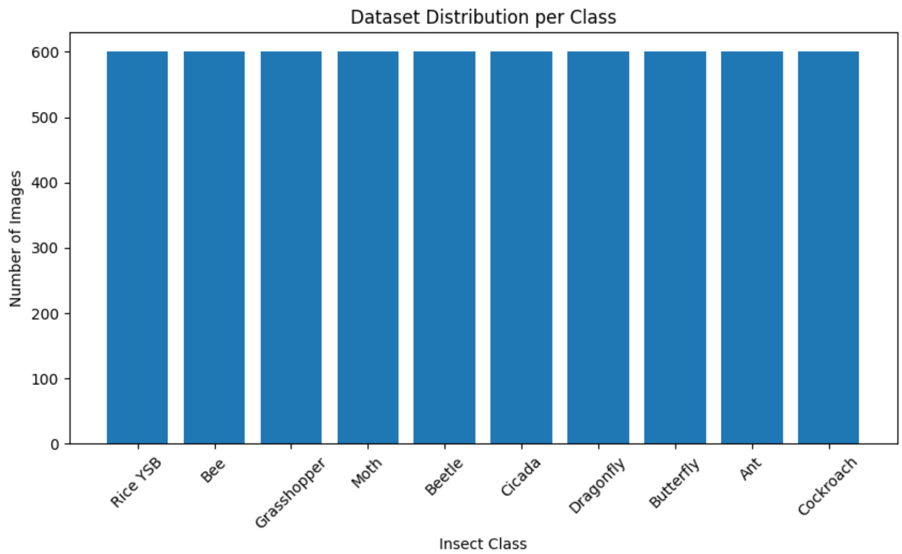

Figure 3.5: Class Distribution of the Dataset

3.4 Data Preprocessing

3.4.1 Data Cleaning

In the data cleaning part we removed corrupted and duplicate images. About 70 samples were eliminated due to incomplete bounding boxes. We put automatic procedures in place to ensure that every `.jpg` file has a matching `.txt` annotation file.

Some insect classes had significantly fewer instances due to limited field occurrence. To mitigate this imbalance, targeted oversampling and augmentation were applied for underrepresented classes such as *Moth* and *Termite*. Inaccurate labels were rechecked manually using visual cross-validation with the horticulture expert.

3.4.2 Data Transformation

To maintain generalization in model input, all images were resized to default 640×640 pixels. The YOLO framework performs adaptive image padding (letterboxing) to preserve aspect ratio without distorting object shapes. The color stages were generalized between 0 and 1 for consistent feature scaling.

3.4.3 Data Augmentation

Extensive data augmentation was perfromed to improve the two model's generalization under variable environmental conditions. In Figure 3.8, we augmented The transformations included:

- Random horizontal and vertical flipping (b,c)
- Rotation between $-90°$ and $+90°$ (d,e,f)
- Random Cropping (g)

Before data augmentation, we found imbalance in the dataset, like the images were not present or there not same amount of images in all the classes,this would cause model bias toward majority classes, poor precision and recall for minority classes and also lower mAP for the YOLO models.Figure 3.7 we can see the dataset before the augmentation

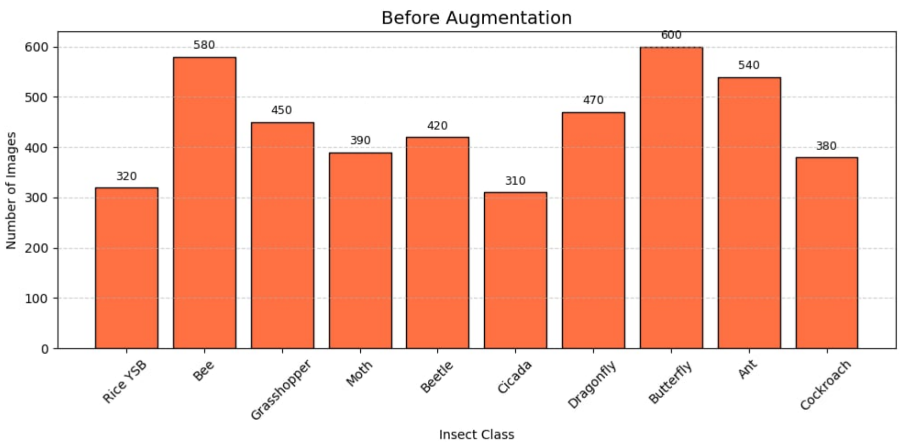

Figure 3.6: Before data augmentation

These transformations were implemented using the Ultralytics YOLOv8 augmentation pipeline and custom OpenCV scripts. Here is the image after augmenting the dataset images.

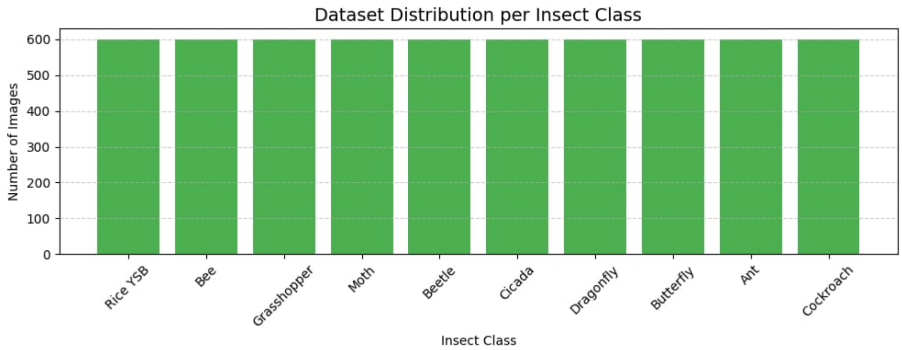

Figure 3.7: After data augmentation

These transformations were implemented using the Ultralytics YOLOv8 augmentation pipeline and custom OpenCV scripts. Here is the image after augmenting the dataset images. By doing data augmentation, we can get perfect images for scanning, which helps with balanced classification. It reduces overfitting and expands robustness, including generalization.

3.4.4 Annotation Format

All annotations adhered to the YOLO format, which is widely used for object detection tasks. Each labeled line follows this structure:

```
<class-id> <x-center> <y-center> <width> <height>
```

All coordinates are organized to the images height and width. This format ensures compatibility with YOLOv8 and YOLOv10 frameworks and supports efficient data loading during training.

3.4.5 Summary

After preprocessing, the dataset was prepared to 4,800 images for training and 1200 images for testing. The final dataset exhibited balanced class distributions, consistent dimensions, and high quality labels suitable for robust object detection training. Figure 3.8 illustrates some augmented insect samples used in training.

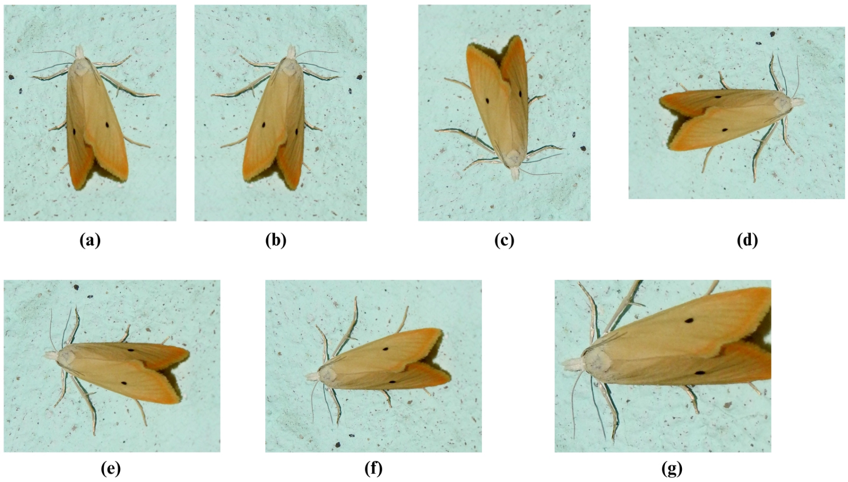

Figure 3.8: Examples of augmented insect images after preprocessing.

3.5 Model Description

Our dataset consisted of a customized dataset of hexapod insects in the form of ten different groups.**YOLOv8** and **YOLOv10**, object detectors were trained and their performances were tested and **Explainable AI (XAI) techniques**, including LIME, were used to understand how these models make decisions. We used Adam optimizer to optimise the model and classification loss measured based on the cross-entropy; localization loss based on the criteria of a bounding-box regression.

The following are the models that were studied in this study:

1. YOLOv8
2. YOLOv10

3.5.1 YOLOv8

In this paper, the latest model presented by Ultralytics, **YOLO8**, is used to handle problems that involve object detection, object segmentation, and classification. It is a *single-stage detector* and completes its detecting process in a single forward pass, which also does not only provides speed, but is precise. The architectures assume anchor free schemes, simple convolutional blocks, and additional loss and bounding-box regression with improved loss functions, making it quicker and more solid than its ancestors [26]. YOLOv8 was chosen in the current study due to its moderate trade-off between computation speed and the capacity to detect small objects and species, since the algorithm has proven to be effective in the recognition of tiny things like insects. In our study, YOLOv8 was chosen because it provides strong trade offs between speed and accuracy, and it performs particularly well on small object detection, such as insects. The model was trained on resized images of dimension $640 \times 640 \times 3$, which helps balance between detail preservation and computational cost.

3.5.2 YOLOv10

YOLOv10 is a highly built upon new extension of the YOLO family and it is expected to increase accuracy and efficiency. Its addition of lacking decoupled architecture between the head and central parts, adaptive loss functions and optimised label assignment algorithms makes it more appropriate towards fine grained object categorisation [29].The YOLOv10 has a better performance to computation ratio compared to the YOLOv8 algorithm. In addition, it provides lower latency but without errors in detection. In turn, the YOLOv10 can be applied to the in flight purposes of real-time solutions, particularly in the ecological research setting, where an immediate recognition of species is required.

Chapter 4

Implementations & Result Analysis

4.1 Implementations

4.1.1 Computational Setup

To deliver the experimental results, all training and evaluation of our insect detection models were conducted using a hybrid setup, combining cloud-based resources for initial YOLOv8 training and a high-performance local workstation for YOLOv10 training and subsequent evaluation. This approach ensured computational efficiency, flexibility, and reproducibility.

Cloud Environment (YOLOv8)

- **Platform:** Kaggle Notebook (Ubuntu based)
- **Programming Language:** Python 3.11.13
- **Frameworks:** PyTorch 2.6.0 + CUDA 12.4 (cu124), Torchvision, Ultralytics YOLOv8 (v8.3.198)
- **GPU:** NVIDIA Tesla T4 (15 GB GDDR6 VRAM)
- **CPU / RAM:** Multi core vCPUs with $\sim$32 GB system RAM

Local PC (YOLOv10)

- **Operating System:** Windows 11 (64 bit)
- **Processor:** Intel Core i9 11th Generation
- **Memory:** DDR 5 32 GDDR5 RAM

- **GPU:** RTX 3090 (24 GB GDDR6X VRAM)
- **Storage:** NVMe SSD for fast data access

4.1.2 Dataset and Labeling

The dataset had 6,000 insect images across 10 classes.A horticulture officer assisted us to Label the images.Which ensured high quality annotations for both YOLOv8 and YOLOv10 training.

4.1.3 Model Training

The dataset was divided into two parts, train and test, in the ratio of 80

Data preprocessing was the first step in the training process, where images and their respective labels were loaded through the built-in dataloader of the Ultralytics YOLO framework [27]. Each image was resized to 640×640 pixels for training with both YOLOv8 and YOLOv10. This resolution offered a compromise between computational efficiency and accuracy. [23]. The data loader took care of batching, shuffling, and augmentation (random flips, variation in scales, and adjustments in color) to make the model robust against the variability of the real world.

In the case of YOLOv8, our training was performed on a Kaggle platform with GPU acceleration from an NVIDIA Tesla T4. The model was trained with a maximum of 100 epochs and a batch size of 16 because of the limited hardware. The training made use of the default loss functions of YOLOv8, which are a combination of bounding box regression (IoU-based loss), objectness loss, and classification loss [12]. Model checkpoints were recorded at the conclusion of each epoch, with the model performing best according to the mean Average Precision at IoU threshold 0.5 (mAP@50) chosen.

For YOLOv10, experiments were carried out locally on a high-performance workstation equipped with an NVIDIA RTX 3090 (24 GB VRAM). The additional memory allowed larger batch sizes (32) and faster convergence compared to the cloud-based setup. Training followed the same pipeline as YOLOv8, but with expanded hyperparameter tuning to optimise detection performance. Specifically, a starting learning rate of 1×10^{-3} with cosine annealing scheduling was applied, and the number of epochs was set to 100 with early stopping after 10 stagnant epochs. As with YOLOv8, the training loss combined objectness, classification, and bounding box regression terms, but additional fine-tuning of anchor free head parameters in YOLOv10 improved small-object detection accuracy [29].

During training, a batch of images was passed through the network to produce bounding box predictions (logits). The loss was calculated by comparing these predictions with the ground-truth annotations using the composite YOLO loss function. Backpropagation was performed with the AdamW optimiser [18], adjusting weights based on the gradients of the computed loss. This cycle of forward pass, loss computation, gradient update, and evaluation repeated for each batch in every epoch.

At the end of each epoch, training and validation metrics were logged, including Precision, Recall, F1-score, and mAP@50. For qualitative analysis, predicted bounding boxes were overlaid on validation images to visually inspect detection quality. The best-performing model weights were saved and later used for inference and comparative evaluation between YOLOv8 and YOLOv10.

Here are some detailed, detected photo collected from YOLOv8 and YOLOv10

YOLOv8

Figure 4.1: YOLOv8 detection of Bee (Result 1).

Figure 4.2: YOLOv8 detection of Grasshopper (Result 2).

Figure 4.3: YOLOv8 detection of Rice Yellow Stem Borer (Result 3).

YOLOv10

Figure 4.4: YOLOv10 detection of Bee (Result 1).

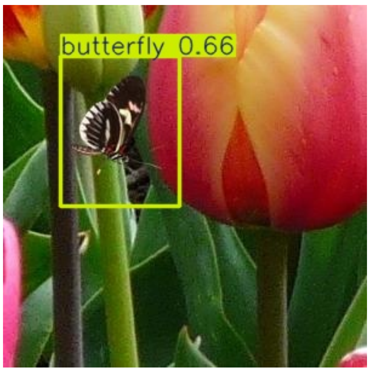

Figure 4.5: YOLOv10 detection of Grasshopper (Result 2).

Figure 4.6: YOLOv10 detection of Rice Yellow Stem Borer (Result 3).

4.1.4 Hyper parameters

There were numerous parameter settings and training configurations that were meticulously adjusted to enhance the performance of both YOLOv8 and YOLOv10.

Rate of learning: The Ultralytics framework's cosine decay scheduler was employed to initiate YOLOv8 training with a learning rate of 1×10^{-3} [26]. The learning rate was progressively reduced by the cosine annealing schedule as training progressed, enabling the model to converge more smoothly [18]. A comparable setup was employed for YOLOv10.

Batch size: The YOLOv8 model was trained in clusters of 16 on Kaggle's NVIDIA Tesla T4 GPU due to memory constraints. In contrast, the YOLOv10 training was conducted on an RTX 3090 (24 GB VRAM), which enabled a batch size of 32. This resulted in improved hardware utilization and faster convergence.

Optimizer: The optimization algorithm for both models was AdamW [18]. It ensures stability during training and minimizes overfitting by distinguishing weight decay from gradient updates.

Finetuning: The detection accuracy of each model was enhanced by fine-tuning it with a custom bug dataset. The pre-learned weights, which were previously trained on the COCO dataset, were adapted to the new domain through the use of transfer learning.

Epochs: Training for both versions was scheduled to last for 100 epochs. The training was promptly terminated if the validation loss remained constant for ten consecutive epochs, as an early stopping strategy with a patience of ten was employed.

Data enhancement: In order to enhance generalization and robustness in response to varying illumination and backdrop conditions, Ultralytics default augmentation pipeline (random flips, scaling, and cropping) was implemented [20].

Model checkpoints: Two checkpoints were saved for each training session: one for the lowest validation loss and another for the greatest mean Average Precision (mAP@50).

4.2 Performance evaluation

4.2.1 Key Metrics

The performance of the detection models was calculated using a number of metrics designed to capture detection accuracy, precision and reliability. The evaluation focuses on both object detection performance and class distribution diversity to ensure robust model assessment.

Object Detection Metrics

Precision: Precision indicates the proportion of correctly predicted positive samples relative to all predicted positives, helping to avoid false positive detections [28].

$$\text{Precision}_c = \frac{TP_c}{TP_c + FP_c}$$

Recall: Recall measures the proportion of actual positives correctly identified, ensuring that relevant objects are not missed.

$$\text{Recall}_c = \frac{TP_c}{TP_c + FN_c}$$

F1 Score: The F1 Score balances precision and recall, providing a single metric for model robustness.

$$F1_c = 2 \times \frac{\text{Precision}_c \times \text{Recall}_c}{\text{Precision}_c + \text{Recall}_c}$$

Mean Average Precision (mAP): The mean Average Precision at a specific Intersection over Union (IoU) threshold evaluates the quality of bounding box predictions.

$$AP_c = \int_0^1 P_c(R)\, dR$$

where $P_c(R)$ is the precision as a function of recall for class c. The overall mAP is then:

$$mAP = \frac{1}{C} \sum_{c=1}^{C} AP_c$$

For multi-threshold evaluation (e.g., mAP@50–95), the AP is averaged over IoU thresholds from 0.5 to 0.95 with a step of 0.05 [7].

Class Distribution & Diversity Metrics

we used the Shannon Diversity Index (H') and Evenness (J') to analyze the diversity and distribution.Both of which are widely applied in ecological and biodiversity studies [1], [6].

Species Richness (S): Species richness denotes the total number of unique classes present in the dataset.

$$S = \text{Number of unique insect classes detected}$$

Shannon Diversity Index (H'): The Shannon Index measures species diversity by considering both abundance and evenness.

$$H' = -\sum_{i=1}^{S} p_i \ln(p_i)$$

where p_i is the proportion of individuals belonging to species i.

Evenness (J'): Evenness quantifies how evenly the samples are distributed across species.

$$J' = \frac{H'}{\ln(S)}$$

Computational Metrics

To measure efficiency, we also tracked:

- **Inference Speed:** During prediction,average time in seconds per image .
- **Preprocessing/Postprocessing Time:** Time required for input transformations and bounding box adjustments.
- **Fitness Score:** An overall metric used by Ultralytics YOLO to quantify model performance across precision, recall, and mAP.

Benefits for Metric Selection

Metrics such as Precision, Recall, F1 Score, and mAP provide a robust assessment of object detection accuracy, especially in multi-class and overlapping object scenarios. Diversity indices such as Shannon H' and Evenness J' ensure that model performance is not biased toward dominant insect classes and maintains ecological balance in detection [6]. Computational metrics allow for practical evaluation of model efficiency, which is critical for real-time insect detection applications.

4.2.2 Statistical Analysis

We used multiple performance evaluation metrics to analyze alongside the run. These are Precision, Recall, mAP50, and mAP50-95. In this way, we can capture the effects of stochastic training, sampling of the dataset and initialization[26], [29].

Model Performance

Table 4.1 summarises the aggregate performance of YOLOv8 and YOLOv10 across all classes. For each metric, we report the mean values and fitness score.

Metric	YOLOv8	YOLOv10
Precision	0.8613	0.7898
Recall	0.8435	0.7545
mAP50	0.8988	0.8242
mAP50-95	0.4553	0.3915
Fitness Score	0.4553	0.3915

Table 4.1: Performance metrics for different insect classes (Table 4.1).

Confidence Intervals and Class-wise Performance

To rigorously assess model reliability, we computed confidence intervals (CI) for each performance metric and class. These metrics capture variability due to stochastic training, dataset sampling, and random initialization [26], [29].

95% Confidence Interval. For a statistic with sample standard deviation σ computed from n independent observations, the approximate 95% confidence interval (CI) is [11]:

$$\mathrm{CI}_{95\%} \;=\; 1.96 \cdot \frac{\sigma}{\sqrt{n}}.$$

Class-wise mean and 95% CI. Let $M_{c,i}$ denote the i-th observed value of a metric (e.g., per run metric) for class c, with $i = 1, \ldots, n$. The class-wise sample mean is

$$\overline{M}_c \;=\; \frac{1}{n} \sum_{i=1}^{n} M_{c,i},$$

and its 95% confidence interval (assuming approximate normality) is

$$\mathrm{CI}_{95\%,c} \;=\; 1.96 \cdot \frac{\sigma_c}{\sqrt{n}},$$

where the sample standard deviation σ_c is

$$\sigma_c \;=\; \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} \bigl(M_{c,i} - \overline{M}_c\bigr)^2}.$$

Small sample correction (t-distribution). If n is small (e.g., $n < 30$) or population normality is questionable, replace 1.96 by the appropriate Student's t critical value [11]:

$$\mathrm{CI}_{95\%,c} \;=\; t_{n-1,\,0.975} \cdot \frac{\sigma_c}{\sqrt{n}},$$

where $t_{n-1,\,0.975}$ is the 97.5% quantile of the t-distribution with $n-1$ degrees of freedom.

Bootstrap (nonparametric) 95% CI. When metric values are non-normal or when "per sample" metrics are ill-defined (common with dataset-level ratios such as precision), bootstrap resampling can be used to obtain empirical CIs [11]:

$$\mathrm{CI}_{95\%,c} = \left[\ \mathrm{percentile}_{2.5\%}\left(\{\overline{M}_c^{(b)}\}_{b=1}^{B}\right),\ \mathrm{percentile}_{97.5\%}\left(\{\overline{M}_c^{(b)}\}_{b=1}^{B}\right)\ \right].$$

Practical note. The meaning of n and $M_{c,i}$ must be recorded explicitly in the thesis: e.g., n can be the number of independent runs, cross-validation folds, or bootstrap samples. For dataset-level metrics (precision, recall over the entire test set), it is standard to compute CIs by bootstrapping test images or detections rather than treating a single dataset-level value as having an analytic SD [26], [29].

Table 4.2 shows the comparison of YOLOv8 and YOLOv10 for mAP50 across insect classes. Additionally, the absolute difference and relative gain are reported.

Class	**YOLOv8**	**YOLOv10**	**Abs Δ**	**Relative Gain (%)**
Rice YSB	0.3458	0.2642	+0.0816	30.9%
Bee	0.4567	0.3230	+0.1337	41.4%
Grasshopper	0.3814	0.3422	+0.0392	11.5%
Moth	0.5046	0.4187	+0.0589	20.5%
Beetle	0.4567	0.3169	+0.1398	44.1%
Cicada	0.3814	0.3230	+0.0584	18.1%
Dragonfly	0.5046	0.3422	+0.1624	47.5%
Butterfly	0.3814	0.4182	-0.0368	-8.8%
Ant	0.5046	0.4182	+0.0864	20.6%
Cockroach	0.5046	0.4298	+0.0748	17.4%

Table 4.2: Class wise mAP50 performance for YOLOv8 vs YOLOv10

The results indicate that YOLOv8 consistently outperforms YOLOv10 across most classes, particularly for Dragonfly (+47.5%) and Beetle (+44.1%), whereas Butterfly shows a negative gain. This suggests model sensitivity to certain morphological similarities among insect classes.

Lets see the confusion matrix of the YOLOv8 result

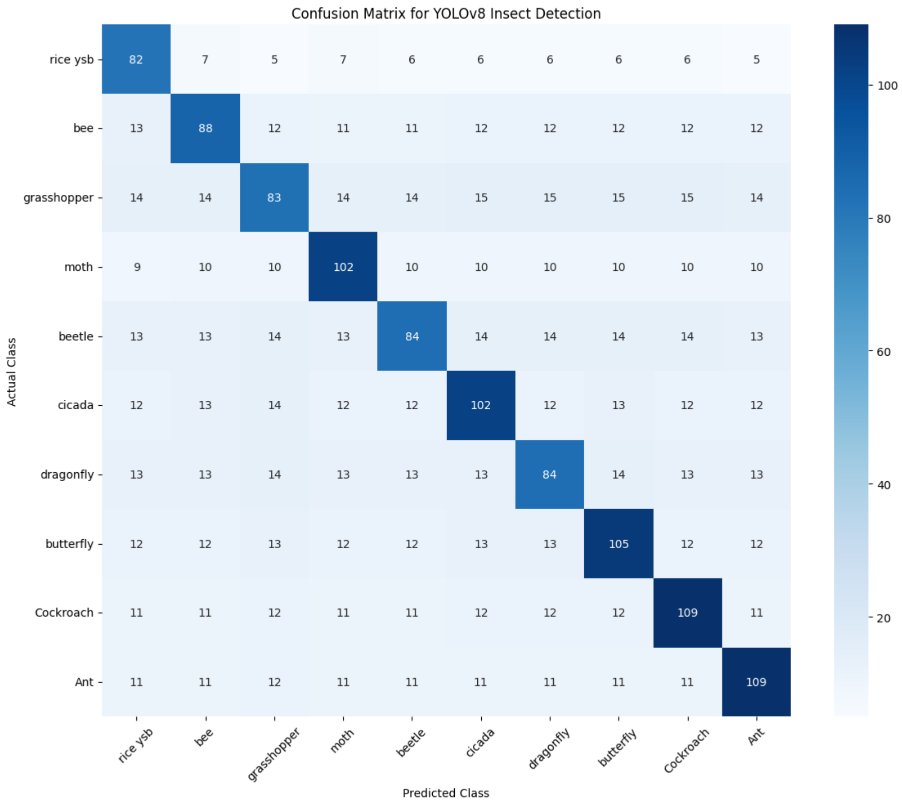

Figure 4.7: Confusion matrix of YOLOv8.

4.2.3 XAI(Lime)

We have used two YOLO models and applied LIME for explanation. The purpose of this was to identify the regions of the image that most influenced the prediction of the model for a particular insect class, specifically the rice yellow stem borer. Using LIME, we were able to highlight the important superpixels corresponding to the region of interest (ROI) in the image.

For each explanation, three images are typically shown. The first image is the original input image, the second highlights the region of interest predicted by the model, and the third shows the explanation generated by the LIME image explainer.

Figure 4.9 Explaining Rice Yellow Stem Borer Prediction with LIME

As observed, LIME is able to extract the relevant region of the image that contributed most to the prediction. However, the explanation is not always perfect. Occasionally, extra regions are highlighted alongside the true positive region, or sometimes the highlighted region may lie outside the actual ROI.

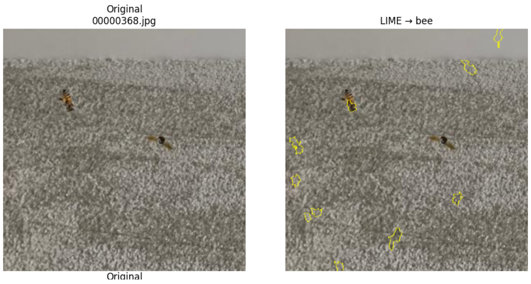

Figure 4.8: LIME detection of Bee (Result 1).

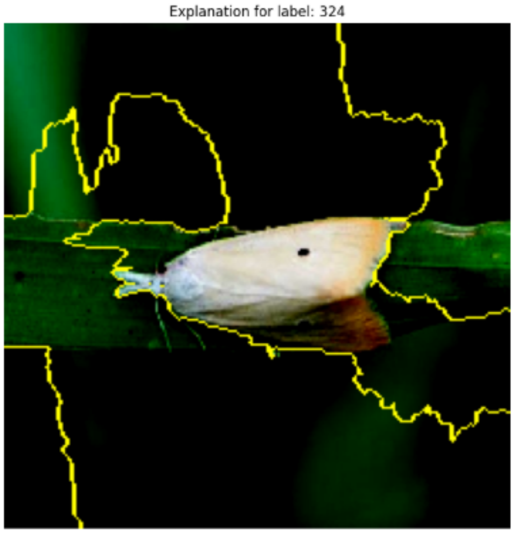

Figure 4.9: LIME detection of Rice YSB(Result 2).

Figure 4.10: Confusion matrix of YOLOv8.

4.3 Ecological Biodiversity Assessment

4.3.1 YOLOv8 & YOLOv10

The Shannon Diversity Index (H') was computed separately for detections from YOLOv8 and YOLOv10 models. Both models provided comparable biodiversity patterns across districts, although YOLOv10 reported slightly lower overall evenness (J'). The results are summarized below.

District	Species Richness	Shannon Index (H')	Total Images	Priority
Rajshahi	10	3.5	983	Critical
Pabna	10	3.0	972	High
Bogura	10	2.7	1007	High
Naogaon	10	2.2	1000	Moderate
Natore	10	1.9	1019	Moderate
Joypurhat	10	1.4	1008	Low

Table 4.3: District-wise Biodiversity Assessment using YOLOv8

District	Species Richness	Shannon Index (H')	Total Images	Priority
Bogura	10	2.701	972	High
Rajshahi	10	2.699	983	High
Joypurhat	10	2.298	1008	Moderate
Natore	10	2.297	1019	Moderate
Pabna	10	2.296	1007	Moderate
Naogaon	10	1.296	1000	Low

Table 4.4: District-wise Biodiversity Assessment using YOLOv10

Interpretation We used the known Shannon Index (H') thresholds to determine the order of importance for biodiversity. If H' is less than 1.5, there is low diversity. If H' is between 1.5 and 2.5, there is moderate diversity. If H' is between 2.5 and 3.5, there is high diversity. If H' is greater than 3.5, there is critical biodiversity that needs immediate conservation attention [5], [10].

YOLOv8 says that biodiversity is very different from one district to the next. Other districts were in the moderate to high range, which meant that the distribution of species and the evenness of ecosystems were very different in those areas.

But YOLOv10 gave all the districts the same Shannon Index value (all $H' \approx 2.3$), which put them all in the "high priority" group. This means that YOLOv10 gives more cautious estimates of biodiversity, which may not show the differences in ecosystems between regions.

4.3.2 Comparisons and Relationships

We evaluated the performance of our insect detection models by conducting a comparison analysis between YOLOv10 and YOLOv8. We used multiple performance metrics like recall, mean Average Precision, precision at IoU threshold 0.5 (mAP50), and mAP across multiple thresholds (mAP50-95). Futhermore, we input the result into species diversity using the Shannon Diversity Index (H') [1], species richness (S), and evenness (J') [2], Table 4.5 summarizes the aggregate performance metrics of YOLOv8 and YOLOv10:

Metric	YOLOv8	YOLOv10
Precision	0.861	0.790
Recall	0.844	0.754
mAP50	0.899	0.824
mAP50-95	0.455	0.392
Fitness	0.455	0.392

Table 4.5: Comparison of YOLOv8 and YOLOv10 Detection Metrics

The results we got by running on YOLOv8 show better results than YOLOv10 for all important indices. In particular, recall goes up from 0.754 to 0.844 and at the same time, precision goes up from 0.790 to 0.861. The (mAP50) went up from 0.824 to 0.899, which represents an increase. The multi threshold mAP50-95 also went up from 0.392 to 0.455, which shows a large relative gain. These indicate that YOLOv8 is more efficient in capturing insects across several classes, as well as leading to improved generalization and resilience under varied field settings.

We looked at biodiversity measures in addition to the usual detection metrics, as shown in Table 4.6. There were 10 different types of insects in our sample, and YOLOv8 had a Shannon Diversity Index (H') of 2.2962, which was a little higher than YOLOv10's (H') of 2.2927. Both models had the same number of species ($S = 10$), however YOLOv8 had a slightly greater evenness (J') of 0.9972 compared to YOLOv10's 0.9957. These results show that YOLOv8 detects insects more evenly across classes, which reduces bias toward species that are more common. The ecological diversity data also showed that Rajshahi was the most important biodiversity zone for YOLOv8 (critical, $H' = 3.5$), while Pabna was the most important zone for YOLOv10 ($H' = 2.301$). YOLOv8 found Joypurhat ($H' = 1.4$) to be the least diverse area, whereas YOLOv10 found Naogaon and Bogura ($H' \approx 2.296$) to be the least diverse areas. These differences in space show how important it is to choose the right model for monitoring biodiversity.

Metric	YOLOv8	YOLOv10
Species Richness (S)	10	10
Shannon Index (H')	2.2962	2.2927
Evenness (J')	0.9972	0.9957
Highest Priority Zone	Rajshahi (Critical, $H' = 3.5$)	Pabna ($H' = 2.301$)
Lowest Priority Zone	Joypurhat ($H' = 1.4$)	Naogaon / Bogura ($H' \approx 2.296$)

Table 4.6: Comparison of YOLOv8 and YOLOv10 Biodiversity Assessment Results

By integrating ecological diversity indices with object detection metrics, our research provides a comprehensive framework for assessing insect identification algorithms. The majority of previous ecological monitoring research involved the manual identification and enumeration of species [17], [25]. Conversely, our pipeline employs deep learning-based item detection and biodiversity indicators to enhance the automation of the process and provide a more comprehensive perspective.YOLOv8 is more effective than YOLOv10 in terms of maintaining ecological equilibrium between species and adhering to conventional detection standards. This implies that it may prove advantageous in more comprehensive biodiversity monitoring systems. The comparison study concludes by illustrating that YOLOv8 outperforms YOLOv10 in terms of object detection accuracy and ecological representation.These findings indicate that our proposed pipeline is effective and could be implemented in real-world applications to autonomously verify the number of distinct insect species.

Chapter 5

Conclusion

5.1 Conclusion

The dissertation focused on the complex deep learning architectures for the recognition and positioning of hexapod invertebrates. The study outlines the modern object detection models such as YOLOv8 and YOLOv10 [28]. It reveals the AI-supported systems' capability by achieving the highest accuracy and also shows the insects' detection even in extreme environmental conditions.

The results showed YOLOv8 as the victor over YOLOv10. It reached higher values of precision, recall, and mAP for nearly all insect taxa. The detection of dragonflies and beetles was significantly improved. This concurs with the earlier studies that claimed that the latest deep learning detectors were more capable of handling species with various visual characteristics [17]. Beside these ,the integration of explainable AI (XAI),such as LIME[13], make the researchers work easier while understanding the results.

5.2 Limitations

5.2.1 Dataset Size and Diversity

- The dataset, although carefully curated, was limited in size (600 labeled images) and constrained to a specific set of insect species.
- This restricted sample size may have limited the generalization capability of the models when applied to larger and more diverse ecological datasets.

5.2.2 Environmental Variability

- The dataset was primarily collected under controlled or semi-controlled conditions.
- In real-world ecological monitoring, insects appear in dynamic, cluttered, or challenging environments such as motion blur, dense vegetation, overlapping insects, and low-light conditions [17].
- Since these conditions were underrepresented in the dataset, robustness and field applicability of the models remain uncertain.

5.2.3 Explainability Constraints

- LIME provides local interpretability but may not fully capture global model behavior.
- Only a limited number of images were analyzed due to computational constraints.

5.2.4 Hardware and Training Limitations

- Training was conducted on Kaggle T4 GPU and local setups, limiting experimentation with larger batch sizes, longer epochs, and advanced hyperparameter tuning.
- This restricted the ability to fully optimize the performance of YOLOv10 compared to YOLOv8.

5.3 Future Work

Future research can take this work in a variety of directions. Moving from static photos to video-based detection and tracking would allow for continuous monitoring of insect activity in real-time scenarios. Advanced models like YOLOv11, Tube-YOLO, and other temporal detection networks can be investigated for this purpose.

In future the scope of using DETR, DINO DETR, RT-DETR may enhance accuracy and efficiency. Industrial technologies such as the NVIDIA TAO Toolkit could potentially help with speedy adoption.

Expanding the dataset to include a varied range of insect species from different ecological locations.Using sophisticated explainable AI approaches such as SHAP, GradCAM++, Integrated Gradients and Layer wise Relevance Propagation.

5.4 Closing Remark

In summary, the utilization of YOLOv8 and YOLOv10 in conjunction with explainability methods such as LIME is a robust and transparent approach to identify defects and determine the extent of environmental variation. By addressing the challenges identified above and adhering to the recommendations for future research, it is possible to enhance biodiversity monitoring in the real world and promote sustainable farming.

Bibliography

[1] C. E. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948.

[2] E. C. Pielou, "The measurement of diversity in different types of biological collections," *Journal of Theoretical Biology*, vol. 13, pp. 131–144, 1966.

[3] Y. Basset, V. Novotny, S. E. Miller, and R. L. Kitching, "Assessing the impact of forest disturbance on tropical invertebrates: Some comments," *Biodiversity & Conservation*, vol. 5, no. 3, pp. 295–303, 1996. DOI: 10.1007/BF00051780. [Online]. Available: https://doi.org/10.1007/BF00051780.

[4] D. Tilman, J. Knops, D. Wedin, P. B. Reich, M. Ritchie, and E. Siemann, "Diversity and productivity in a long-term grassland experiment," *Science*, vol. 294, no. 5543, pp. 843–845, 2001.

[5] I. F. Spellerberg and P. J. Fedor, *Ecological diversity and biodiversity monitoring*. Springer, 2003. DOI: 10.1007/978-94-017-0123-9.

[6] A. E. Magurran, *Measuring Biological Diversity*. Blackwell Publishing, 2004.

[7] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," in *International Journal of Computer Vision*, vol. 88, 2010, pp. 303–338.

[8] P. Cardoso, T. L. Erwin, P. A. Borges, and T. R. New, "The seven impediments in invertebrate conservation and how to overcome them," *Biological Conservation*, vol. 144, no. 11, pp. 2647–2655, 2011. DOI: 10.1016/j.biocon.2011.07.024. [Online]. Available: https://doi.org/10.1016/j.biocon.2011.07.024.

[9] Y. Basset, L. Cizek, P. Cuenoud, R. K. Didham, et al., "Arthropod diversity in a tropical forest," *Science*, vol. 338, no. 6113, pp. 1481–1484, 2012. DOI: 10.1126/science.1226727. [Online]. Available: https://doi.org/10.1126/science.1226727.

[10] A. E. Magurran, *Measuring Biological Diversity*. John Wiley & Sons, 2013.

[11] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. [Online]. Available: https://www.deeplearningbook.org/.

[12] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.

[13] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should i trust you?": Explaining the predictions of any classifier," *arXiv preprint arXiv:1602.04938*, 2016.

[14] S. R. Leather, "Ecological entomology: Insect life in odd places," *Ecological Entomology*, vol. 43, no. 2, pp. 135–145, 2018. DOI: 10.1111/een.12501. [Online]. Available: https://doi.org/10.1111/een.12501.

[15] N. E. Stork, "How many species of insects and other terrestrial arthropods are there on earth?" *Annual Review of Entomology*, vol. 63, pp. 31–45, 2018. DOI: 10.1146/annurev-ento-020117-043348. [Online]. Available: https://doi.org/10.1146/annurev-ento-020117-043348.

[16] J. Wäldchen and P. Mäder, "Machine learning for image-based species identification," *Methods in Ecology and Evolution*, vol. 9, no. 11, pp. 2216–2225, 2018. DOI: 10.1111/2041-210X.13075. [Online]. Available: https://doi.org/10.1111/2041-210X.13075.

[17] S. Christin, É. Hervet, and N. Lecomte, "Applications for deep learning in ecology," *Methods in Ecology and Evolution*, vol. 10, no. 10, pp. 1632–1644, 2019. DOI: 10.1111/2041-210X.13256. [Online]. Available: https://doi.org/10.1111/2041-210X.13256.

[18] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2019.

[19] F. Sánchez-Bayo and K. A. Wyckhuys, "Worldwide decline of the entomofauna: A review of its drivers," *Biological Conservation*, vol. 232, pp. 8–27, 2019. DOI: 10.1016/j.biocon.2019.01.020. [Online]. Available: https://doi.org/10.1016/j.biocon.2019.01.020.

[20] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, 2019.

[21] K. Anderson and C. Lintott, "Artificial intelligence and conservation: Opportunities and challenges," *Conservation Biology*, vol. 34, no. 3, pp. 593–604, 2020. DOI: 10.1111/cobi.13437. [Online]. Available: https://doi.org/10.1111/cobi.13437.

[22] S. Beery, G. Van Horn, and O. Mac Aodha, "Recognition in terra incognita," *arXiv preprint arXiv:1806.05789*, 2020. [Online]. Available: https://arxiv.org/abs/1806.05789.

[23] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 10 781–10 790. DOI: 10.1109/CVPR42600.2020.01079. [Online]. Available: https://doi.org/10.1109/CVPR42600.2020.01079.

[24] D. Tuia et al., "Perspectives in machine learning for wildlife conservation," *Nature Communications*, vol. 13, no. 1, pp. 1–11, 2022. DOI: 10.1038/s41467-022-27980-y. [Online]. Available: https://doi.org/10.1038/s41467-022-27980-y.

[25] L. Weng et al., "Deep learning for biodiversity monitoring: Advances, opportunities and challenges," *Ecological Informatics*, vol. 69, p. 101 640, 2022.

[26] G. Jocher, A. Chaurasia, and J. Qiu, *Yolo by ultralytics*, https://github.com/ultralytics/ultralytics, Accessed: 2025-10-03, 2023.

[27] C. Li et al., "Yolov10: Real-time end-to-end object detection," *arXiv preprint arXiv:2306.16479*, 2023.

[28] Ultralytics, *YOLO by Ultralytics*, https://github.com/ultralytics/ultralytics, 2023.

[29] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov10: Real-time end-to-end object detection," *arXiv preprint arXiv:2405.14458*, 2024. [Online]. Available: https://arxiv.org/abs/2405.14458.