Enhanced Tree Species Detection in Dense Forest Ecosystems: Leveraging Innovative Techniques and Multi-spectral satellite imagery

Harish Dhatchina Moorthy

Master of Science
Data Science
School of Informatics
University of Edinburgh
2024

Abstract

This thesis investigates the performance of multiple models for tree species detection and health assessment in dense, heterogeneous forests (Forest of Dean). Models explored include traditional algorithms like Random Forest and Neural Networks, Extreme Gradient Boosting alongside advanced architectures such as Faster Region-based Convolutional Neural Network (Faster RCNN). Notably, this work innovatively applies the latest You Only Look Once (YOLO) model, with a particular focus on its YOLOv10x variant, marking its first application in this complex domain. Through comprehensive comparative analysis, the study examines the model's performance across varying hyperparameters (learning rates and epochs) and assesses the impact of data augmentation.We observed, YOLOv10x model exhibited a unique blend of strengths and limitations, particularly within the challenging environment of dense forests. We also proved that pixel-based classification models benefitted significantly from the integration of Vegetation Indices (VI), with the neural network model achieving a remarkable accuracy. And for tree health prediction, the regression model yielded strong results, with Leaf Area Index (LAI) as the health indicator. Finally, the findings emphasize the superior performance of the YOLOv10x in object detection and the Neural network model in pixel-based classification, illustrating the potential for these models to advance the accuracy and efficiency of environmental monitoring in complex forest ecosystems.

Research Ethics Approval

This project was planned in accordance with the Informatics Research Ethics policy. It did not involve any aspects that required approval from the Informatics Research Ethics committee.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Harish Dhatchina Moorthy*)

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Dr. Michael Herrmann, for his unwavering support and guidance throughout this project. His acceptance of my involvement in this meaningful work, coupled with his patience and encouragement, has been invaluable, particularly during moments of frustration and difficulty. Dr. Herrmann consistently guided me in my projects and was always there to assist me, making this process much smoother than it would have been without him.

I would also like to express my sincere gratitude to Juan Suarez-Minguez, Project manager at the Edinburgh Forest Research Facility, for his invaluable assistance in providing access to data and collaborating on various aspects of the project. His expertise and insights have greatly contributed to the advancement of our research objectives.

Lastly, I am profoundly grateful to my parents, whose support has been the foundation of my journey. Studying abroad has been challenging, but they have always provided unwavering encouragement and have been my strongest source of strength. Without them, I wouldn't be the person I am today.

Table of Contents

Chapter 1

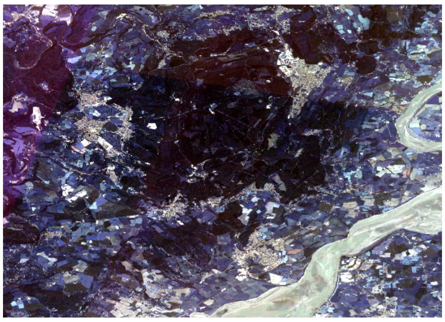
Introduction

1.1 Motivation

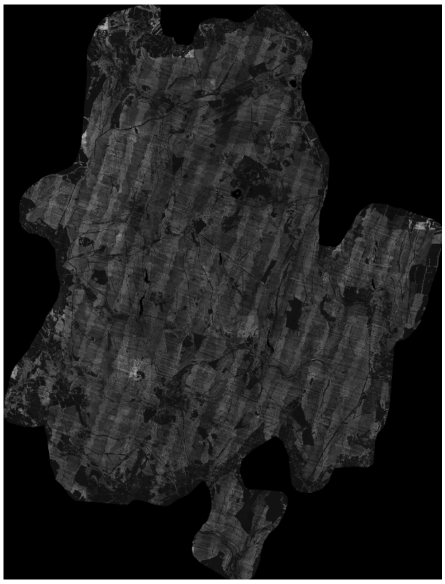
The rapid advancements in artificial intelligence (AI) over the past decades have been significantly shaped by the concurrent evolution of both hardware and software technologies. AI has progressed beyond merely executing predetermined programs, as it now possesses the capability to learn and autonomously adjust its parameters to enhance task performance. This ability to mimic human intelligence has enabled AI to not only perform complex tasks but also to iteratively improve its efficiency based on the data it gathers. Today, AI is being leveraged across a wide array of fields, including computer vision [46], natural language processing (NLP) [9], recommendation systems [37], and beyond, aiding in more informed decision-making processes due to its exceptional performance.

Simultaneously, advancements in remote sensing technologies, particularly in multispectral satellite imagery, have provided unprecedented opportunities for environmental monitoring and resource management. Multispectral imagery, capturing data across various wavelengths, has proven invaluable in fields such as agriculture, forestry, and environmental science [10]. The detailed spectral information allows for the differentiation of various objects and materials on the Earth's surface, including vegetation, soil, and water bodies. Fig 1.1a,1.1b are the spectral images from planet (see section3.2, section 2.4). Among the various applications, the classification of tree species from satellite imagery has emerged as a critical task, given its implications for biodiversity conservation, forest management, and climate change studies [48].

Tree species classification using multispectral satellite imagery presents a unique set of challenges due to the complex spectral signatures of different species and the spatial



(a) Multi-spectral (8-band) imagery from Planet



(b) Uni-spectral (1-band) imagery from Planet, containing LAI scores.

Figure 1.1: SuperDove mosaic images

resolution limitations of satellite data. Traditional models such as Random Forest (RF), Neural Networks, and XGBoost have demonstrated effectiveness in handling large-scale datasets and complex classification tasks, making them suitable for this domain [50]. Additionally, recent advancements in object-based detection techniques, particularly with the advent of models like YOLO (You Only Look Once), have opened new avenues for accurate and efficient tree species identification [12]. YOLO's ability to detect and classify objects within images in real-time offers promising potential for improving the accuracy and speed of species classification from satellite imagery.

Moreover, the health of tree species is an equally critical factor that needs to be addressed in environmental monitoring. Identifying not just the species but also their health status is crucial for assessing forest vitality, monitoring the spread of diseases, and implementing timely conservation strategies. The incorporation of health assessment into species classification adds another layer of complexity to the problem but also increases the relevance and impact of the research. AI-driven methods, particularly those rooted in computer vision, offer a promising solution for simultaneously addressing both species identification and health assessment in tree populations.

Therefore, the motivation behind this research lies in the potential to enhance the accuracy and efficiency of tree species classification and health assessment using a combination of traditional models and modern object-based detection techniques. By applying these AI-driven methodologies to multispectral satellite imagery, this thesis aims to contribute to more effective forest management and conservation efforts, ultimately supporting global biodiversity and environmental sustainability. The utilization of traditional models like RF, Neural Networks, and XGBoost and object-based detection techniques such as YOLO, CNN models offers a novel approach that could significantly advance the field of remote sensing and environmental monitoring.

1.2 Problem Statement

The accurate mapping of tree species within UK forests is a crucial yet challenging task, offering invaluable insights for forest management and conservation efforts. Forest Research has traditionally relied on Sentinel-2 multispectral imagery [15], which is freely available and provides a spatial resolution of 10 x 10 meters. These images, combined with training labels derived from the comprehensive Sub-Compartment Database—containing detailed species information across many forests in England and Scotland—have formed the foundation for tree species classification. However, this pixel-level classification approach, while effective in homogenous, monospecies stands, encounters significant limitations in more complex and heterogeneous forest environments. The coarse spatial resolution of Sentinel-2 imagery often leads to the blending of spectral signatures from different species, undermining the accuracy of the classification, particularly in diverse forest settings [43].

In response to these limitations, the advent of higher resolution multispectral data from Planet Labs' 8-band "SuperDove" Cubesats, with a daily revisit time and an approximately 3.7-meter pixel resolution, presents a promising alternative. The finer spatial scale of SuperDove imagery [40] holds potential for improved species discrimination, especially in heterogeneous forests where individual tree species may occupy smaller areas than the Sentinel-2 pixels can resolve. This advancement necessitates a re-evaluation of current methodologies, comparing pixel-level approaches with object-based detection techniques that could better leverage the increased resolution of SuperDove data. Additionally, alongside species classification, the assessment of tree health within these forests emerges as a critical sub-problem, as accurate health monitoring is essential for effective forest management and conservation efforts.

Given these developments, there is a pressing need to explore the efficacy of high-resolution multispectral data for tree species classification and health detection, particularly in the UK's complex forest landscapes. This exploration is crucial to advancing the current capabilities of remote sensing in forest management, potentially leading to more accurate and actionable insights for forest practitioners .

1.3 Aims and Objectives

This project aims to leverage advanced artificial intelligence techniques to enhance the accuracy and efficiency of tree species classification and health detection in UK forests using multi-spectral satellite imagery. Specifically, it will explore the application of both traditional models (Random Forest, Neural Networks, XGBoost) and object-based detection methods (YOLOv10, Faster RCNN) to classify tree species and assess their health within complex, species-diverse forest landscapes. The research will evaluate the effectiveness of high-resolution data from Planet Labs' "SuperDove" Cubesats in addressing the limitations of existing methodologies that rely on lower resolution Sentinel-2 imagery.

The objectives of this project are: 1. Develop and compare traditional machine learning models and object-based detection techniques for the classification of tree species using multispectral satellite imagery. 2. Assess the potential of high-resolution "SuperDove" data and latest YOLOv10 in improving classification accuracy, particularly in heterogeneous forest environments. 3. Innovatively integrate tree health detection into the classification process to provide comprehensive insights for forest management. 4. Analyze the performance of various AI models and discuss the implications for future advancements in remote sensing-based forest monitoring.

1.4 Key Findings

In this study, we explored various models for tree species detection, applying models like Random Forest, Neural Networks, XGBoost and Faster R-CNN, as well as the innovative YOLOv10 model. To the best of our knowledge, this study marks the first application of YOLOv10 in detecting tree species within dense, heterogeneous forests. While the YOLOv10x model demonstrated high precision, it also exhibited a tendency to predict fewer tree objects due to the effects of confidence and IoU thresholds. Additionally, the integration of Vegetation Indices (VI) in pixel-based classification significantly enhanced model performance, with the neural network model achieving an accuracy of 89%. Finally, the regression model for health prediction showed robust performance with high R^2 values and low error metrics, validating its effectiveness in predicting Leaf Area Index (LAI). Our findings suggest that YOLOv10x, with its adaptability and fine-tuning capabilities, holds the potential to be a leading model in challenging object detection scenarios, despite the conservative prediction approach imposed by its thresholds.

1.5 Structure of the Dissertation

This dissertation is organized into seven chapters. Chapter 2 will review the existing literature on tree species classification using multispectral satellite imagery, focusing on both traditional machine learning models (Random Forest, Neural Networks, XGBoost) and modern object-based detection techniques (YOLO). The chapter will also cover advancements in remote sensing technologies, particularly the use of high-resolution ”SuperDove” data. Chapter 3 will provide detailed information about the datasets utilized in this research, including the Sentinel-2 imagery and the high-resolution SuperDove data, as well as the Sub-Compartment Database for training labels. In Chapter 4, the methodologies employed in this study will be presented, encompassing the models, classification approaches, and evaluation metrics used for both tree species identification and health detection. Chapter 5 will describe the experimental design, followed by a presentation of the results. Chapter 6 will offer a detailed discussion and analysis of the findings, evaluating the effectiveness of different models and data types. Finally, Chapter 7 will conclude the dissertation, summarizing key insights with Chapter 8, proposing directions for future research.

Chapter 2

Literature Review

The classification of tree species and detection of tree health using multispectral satellite imagery involve intricate challenges that intersect remote sensing, machine learning, and environmental science. This chapter provides a comprehensive review of the methodologies and advancements relevant to these tasks, focusing on traditional machine learning models, emerging deep learning architectures, and object detection techniques. We also explore the role of high-resolution imagery and the application of transfer learning to improve classification accuracy and health detection in forest environments.

2.1 Traditional Machine Learning Models

Neural Networks (NN)

Neural Networks (NN) [3], particularly Multi-Layer Perceptrons (MLPs), have been widely used in classification tasks due to their ability to model complex, non-linear relationships [4]. The backpropagation algorithm, introduced by Rumelhart et al. [38], enables the training of multi-layer networks by adjusting weights through gradient descent. This foundational work established NN as a robust tool for various predictive tasks, including those in remote sensing.

In remote sensing, NNs have been employed for classifying vegetation types and tree species from satellite imagery. Hogland et al. [19] applied NN techniques to Landsat imagery for tree species classification, revealing the model's strength in capturing intricate relationships between spectral bands and tree characteristics. The NN's capacity to learn non-linear mappings between input features and target classes is advantageous in differentiating between species with subtle spectral differences.

Despite their effectiveness, traditional NNs often face challenges related to overfitting and computational demands. To address these issues, researchers have explored advanced NN architectures and regularization techniques to enhance model performance and generalization [33]. These advancements have made NN a valuable tool in remote sensing applications, particularly when dealing with high-dimensional and complex datasets.

Random Forest (RF)

Random Forest (RF) is a powerful ensemble learning method that constructs multiple decision trees during training and outputs the mode of the classes (classification) or mean prediction (regression) of the individual trees. Breiman L [8] introduced RF, emphasizing its ability to handle large datasets and manage high dimensionality effectively. RF's robustness to overfitting and its capability to handle noisy data make it particularly suitable for complex classification tasks in remote sensing.

In the context of tree species classification, RF has demonstrated notable success. For instance, Gislason et al. applied RF to land cover classification using satellite imagery and found that it outperformed traditional single decision trees and other classifiers. Their study highlighted RF's efficacy in dealing with the diverse spectral signatures present in remote sensing data, as well as its resilience to data noise and variability [17]. This characteristic is especially useful for classifying tree species in heterogeneous forest environments, where the spectral signatures of different species can overlap and blend.

Clarke et al. [13] extended the application of RF to the classification of tree species in tropical rainforests using Landsat imagery. Their research demonstrated RF's capability to distinguish between species with similar spectral profiles, underscoring its effectiveness in diverse forest conditions. The RF model's performance was attributed to its ensemble approach, which aggregates predictions from multiple trees to improve overall classification accuracy.

XGBoost

XGBoost (Extreme Gradient Boosting) is an ensemble learning technique that builds on the principles of gradient boosting. Chen and Guestrin [11] introduced XGBoost as a scalable and efficient implementation of gradient boosting, incorporating both regularization and optimization techniques to enhance model performance. XGBoost's flexibility and high performance have made it a popular choice for various classification tasks.

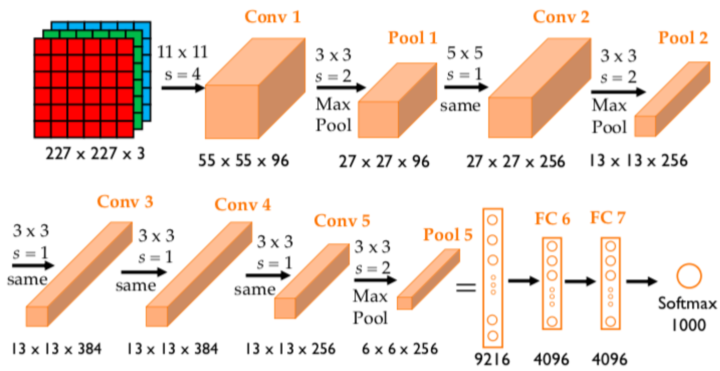


Figure 2.1: Alexnet architecture framework, from Krizhevsky et al. [25].

In the realm of tree species classification, XGBoost has been successfully applied to hyperspectral data. Los et al. utilized XGBoost to classify tree species in a mixed forest using hyperspectral imagery, achieving high classification accuracy of 94%. Their study highlighted XGBoost's ability to handle large feature sets and complex data distributions, making it suitable for remote sensing applications where spectral information is rich and varied [30].

XGBoost's effectiveness is attributed to its boosting mechanism, which sequentially builds trees to correct errors made by previous ones. This approach enhances the model's accuracy and robustness, particularly in scenarios where the data may be imbalanced or contain noise. The integration of XGBoost and Random forest with remote sensing data has proven to be a powerful strategy for improving tree species health detection, with an overall accuracy improved by 16.1% [20].

2.2 Emerging Architectures

Convolutional Neural Networks (CNNs)

Convolutional Neural Networks (CNNs) have revolutionized image classification tasks by automatically learning hierarchical features from raw image data. The seminal work by LeCun et al. [27] on the LeNet architecture demonstrated CNNs' ability to process and classify images effectively. This work laid the foundation for subsequent advancements in CNN architectures, including AlexNet [25] and ResNet [18], which have set new benchmarks in image classification performance . Alexnet architectute is shown in Fig 2.1, which ended up winning second place in ILSVRC-2012 competition.

CNNs have been applied to tree species classification with promising results. Janne

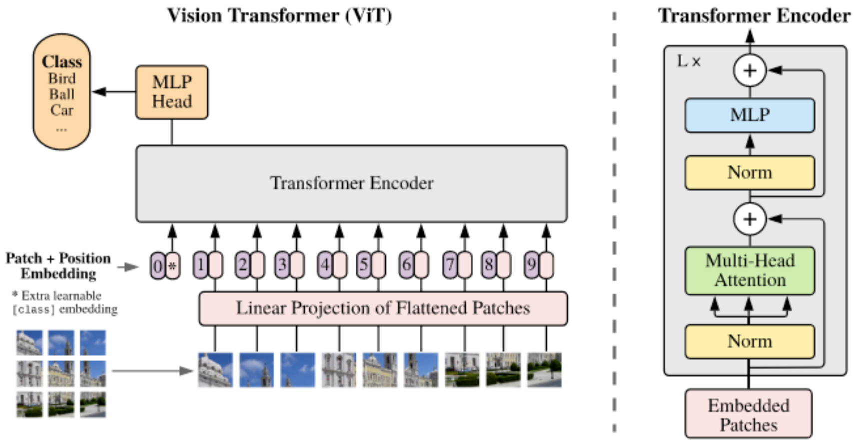


Figure 2.2: Model modified-overview, from Dosovitskiy et al. [14].

Mäyrä et al. used 3D-CNNs to classify tree species from high-resolution aerial imagery with best model having 91% F1-score, showcasing the model's capability to learn complex spatial patterns and spectral features. The CNN's ability to extract detailed features from images allows it to distinguish between tree species with similar spectral signatures [31]. The hierarchical feature extraction process of CNNs enables the model to capture both low-level and high-level features, improving classification accuracy.

CNNs offer several advantages for remote sensing applications, including their ability to handle large-scale data and their robustness to variations in image quality. Faster RCNN is one such model, that obtained the highest F1-score of 94.99%, for oil palm tree detection[52]. However, the training of CNNs can be computationally intensive, and the models require substantial labeled data to achieve optimal performance. Recent advancements in CNN architectures and training techniques continue to address these challenges, making CNNs a powerful tool for tree species classification [51].

Transformer Models

Transformer models, introduced by Vaswani et al. [44], have brought significant advancements to natural language processing and are increasingly being adapted for computer vision tasks. Unlike CNNs, Transformers use self-attention mechanisms to capture long-range dependencies and contextual information, which can be advantageous for complex image classification tasks.

Vision Transformers (ViTs) have also been explored for image classification, with promising results. Dosovitskiy et al. demonstrated the application of ViTs to high-resolution images directly without using self-attention, shown in Fig 2.2, showcasing their ability to handle varying resolutions and complex relationships [14]. The Transformer's ability to process global context and manage large-scale data makes it a compelling choice for tree species classification, particularly in heterogeneous forest environments where traditional methods may struggle.

The integration of Transformers with remote sensing data presents opportunities for enhanced classification accuracy and efficiency. The self-attention mechanism of Transformers allows for better handling of spatial and spectral variations, making them a valuable addition to the toolkit for tree species classification and health detection [6].

2.3 Object-Based Detection Techniques

YOLO (You Only Look Once)

YOLO (You Only Look Once) is a state-of-the-art object detection model introduced by Redmon et al. [36] that processes images in real-time by predicting bounding boxes and class probabilities in a single pass through the network. YOLO's architecture is designed for both speed and accuracy, making it suitable for applications requiring real-time processing and high precision.

In the context of tree species classification and health detection, YOLO's object detection capabilities can be leveraged to identify and classify individual trees within high-resolution imagery. Xiao et al. applied YOLO to aerial imagery for urban tree detection, achieving high accuracy of 79% in identifying tree locations and species. YOLO's ability to handle overlapping objects and varied scales is particularly useful for complex urban-forest environments, where trees may be densely packed or partially obscured [45]. But unfortunately, there are no evidences to them being used in pure forest environments which is surprising, especially when YOLOv8 being the SOTA model for object detection.

YOLO's real-time processing capability allows for efficient analysis of large-scale imagery, making it suitable for monitoring extensive forest areas. YOLOv10 outperforms previous YOLO versions and other state-of-the-art models in terms of accuracy and efficiency [47]. The model's ability to detect and classify objects within images enables detailed analysis of tree species distribution and health status, providing valuable insights for forest management and conservation efforts.

2.4 Benefits of High-Resolution Imagery

The advent of high-resolution multispectral imagery, such as that provided by Planet Labs' "SuperDove" Cubesats, offers significant advantages for tree species classification and health detection. The 3.7-meter pixel resolution of SuperDove data provides finer spatial detail compared to the 10 x 10 meter resolution of Sentinel-2 imagery, allowing for more precise identification of individual trees and their species [42].

High-resolution imagery improves classification performance by enhancing the ability to distinguish between tree species with subtle spectral differences. Siham Acharki [5] demonstrated that high-resolution imagery from drones and satellites led to better classification accuracy compared to lower resolution data with mean absolute error of 6%. The increased spatial detail facilitates more accurate detection of tree health indicators, such as leaf discoloration or canopy density, which are crucial for effective forest management.

The ability to analyze fine-scale features and detect subtle changes in tree health with high-resolution imagery provides valuable insights for monitoring forest vitality and implementing conservation strategies. The integration of high-resolution data with advanced classification and detection techniques holds promise for improving the accuracy and effectiveness of tree species classification and health assessment.

2.5 Transfer Learning in Remote Sensing

Transfer learning involves leveraging pre-trained models on related tasks to enhance performance on new, but similar, tasks. This approach has gained prominence in remote sensing due to the high computational cost of training deep learning models from scratch. Pan et al. [34] provided an overview of transfer learning techniques and their applications, emphasizing their potential for improving model performance with limited labeled data.

In tree species classification, transfer learning can be utilized by fine-tuning pre-trained CNNs or Transformer models on remote sensing datasets. Shi et al. [41] applied transfer learning with CNNs to hyperspectral data, achieving improved classification accuracy of 76.70% by leveraging models pre-trained on large-scale image datasets. This approach reduces the need for extensive labeled data and accelerates the training process and improve low-performance classification models, making it a valuable strategy for applying deep learning to remote sensing tasks[23].

Transfer learning enables researchers to benefit from the knowledge embedded in pre-trained models, facilitating more efficient and accurate classification of tree species and health detection. The application of transfer learning in remote sensing continues to evolve, offering opportunities for enhanced model performance and reduced computational requirements.

2.6 Health Detection in Trees

The detection of tree health involves identifying indicators of disease, stress, or other adverse conditions through the analysis of spectral signatures and spatial features. Techniques for health detection often incorporate both spectral and spatial analysis to assess tree vitality and detect early signs of deterioration [49].

Vegetation indices, such as the Normalized Difference Vegetation Index (NDVI), are commonly used for assessing vegetation health based on red and near-infrared reflectance [21]. NDVI provides a measure of vegetation vigor and can indicate changes in tree health. However, NDVI alone may not be sufficient for detailed health assessment, especially when dealing with complex tree species and varying environmental conditions.

Recent research has explored the use of advanced spectral indices and machine learning models to improve health detection accuracy. Bergmüller et al. [7] integrated spectral indices with machine learning classifiers to enhance the detection of tree diseases, demonstrating improved sensitivity and specificity compared to traditional methods. The combination of spectral data and advanced classification techniques enables more precise assessment of tree health and facilitates timely intervention.

The integration of high-resolution imagery and advanced machine learning models holds promise for improving tree health detection and providing valuable insights for forest management. By leveraging these technologies, researchers can enhance their ability to monitor and manage forest ecosystems effectively.

Chapter 3

Data Collection, Integration, and Feature Extraction

The accuracy and effectiveness of tree species classification and health detection hinge on the quality and comprehensiveness of the data used. This chapter details the area of study , methodologies employed for data collection, integration, and feature extraction in the context of this research. The approach integrates high-resolution multispectral imagery, LiDAR data, field observations, and advanced data processing techniques to build a robust dataset for machine learning and object detection models. And a constant data split of 70% train, 20% validation and 10% test was opted for all the models used.

3.1 Area of study (Forest of Dean)

The Forest of Dean (Fig 3.1), located in Gloucestershire, England, is one of the oldest surviving ancient woodlands in the UK [32]. Spanning over 110 square kilometers, it is a biologically rich area, offering a diverse range of habitats and species. The forest is a mixed woodland, primarily consisting of oak, beech, and sweet chestnut, but also includes a variety of other tree species like birch, ash, Scots pine and so on, totalling upto 14 different species.

The Forest of Dean is managed by Forestry England, and their efforts in sustainable forestry practices contribute to the maintenance of this complex ecosystem. The total number of trees in the Forest of Dean is challenging to quantify due to its extensive area and density, but estimates suggest millions of trees thrive here. This environment's rich biodiversity, alongside human management, offers a compelling case study for applying tree species classification methods, potentially enhancing conservation efforts

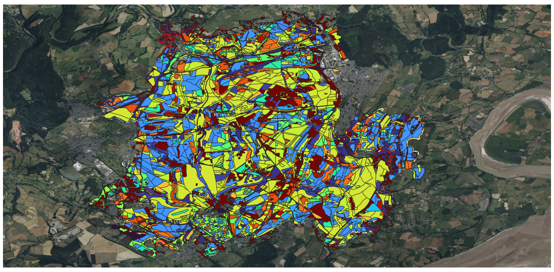


Figure 3.1: Forest of Dean sub-compartment data with colors denoting different species.

and forest management strategies.

3.2 Data Collection

Planet Data Acquisition

To capture the temporal and spectral variability of tree species, high-resolution multi-spectral imagery was obtained from Planet Labs. Planet's SuperDove Cubesats provided the multi-spectral data with a resolution of approximately 3.7 meters per pixel. This high-resolution imagery is crucial for distinguishing between individual trees and capturing fine details necessary for accurate species classification. By collecting these images, the dataset encompasses variations in spectral properties, which can be instrumental in identifying tree species and assessing their health.

LiDAR and Field Data Collection

LiDAR (Light Detection and Ranging) data and field observations were acquired from the UK Forest Department to supplement the multi-spectral imagery. LiDAR data offers detailed three-dimensional information about the forest canopy [29], including tree height, canopy structure, diameter breast height, volume, crown area and topography. This data is essential for creating accurate spatial models and for integrating tree-specific measurements with the multi-spectral imagery.

Field data included detailed measurements of individual trees, including species identification, and canopy characteristics. This comprehensive field dataset serves as ground truth for the machine learning models and provides a basis for assessing the accuracy of the species classification.

3.3 Data Integration and Mapping

GIS Mapping with QGIS

Geographic Information System (GIS) technology was employed to integrate and map the collected data using QGIS, an open-source GIS software [26]. The integration process involved aligning the multispectral imagery, LiDAR data, and field observations to create a comprehensive spatial dataset.

The integration began with converting the northing-easting coordinates from the field data to latitude-longitude coordinates, ensuring precise spatial alignment with the satellite imagery. This conversion was crucial for accurate geo-referencing and subsequent analysis. Each tree in the study area was individually referenced, allowing for detailed spatial mapping and alignment of the various data layers.

Field and LiDAR Data Combination

To create a detailed dataset (of 837731 unique values) for analysis, field data was combined with LiDAR data to produce a comprehensive table containing species names and corresponding spatial coordinates, along with tree characteristics. The same process was followed for generating health detection dataset, with an additional column called LAI being extracted using another 1-band SuperDove imagery, provided by the UK forestry. This integration provided a robust foundation for analyzing tree species distribution and health.

For data points where direct matches between field data and imagery were not available, nearest-neighbor techniques were used to assign species names. This approach ensured that all areas within the study region were covered, even if direct spatial correspondence was lacking. By assigning the nearest species names, the dataset achieved comprehensive coverage, which is essential for accurate classification and modeling. And for outlier detection, Isolation forest algorithm was utilized, as it was particularly developed for forest data due to volume constraints [28].

3.4 Feature Extraction

Raster Value Calculation

Raster value extraction involved calculating spectral values for individual trees from the multispectral imagery. This process facilitated the analysis of changes in tree reflectance and enabled the identification of variations in spectral properties. By extracting raster values, it was possible to analyze how different tree species's reflectance patterns change and how these changes can be leveraged for classification.

Spectral Indices and Reflectance Calculation

In addition to raw spectral values, various spectral indices were calculated to enhance the differentiation between tree species. Notable index included the Normalized Difference Vegetation Index (NDVI). NDVI, calculated from the red and near-infrared bands, is a widely used indicator of vegetation health and biomass [21]. The calculation of this index provided additional features that were instrumental in distinguishing between species and assessing their health.

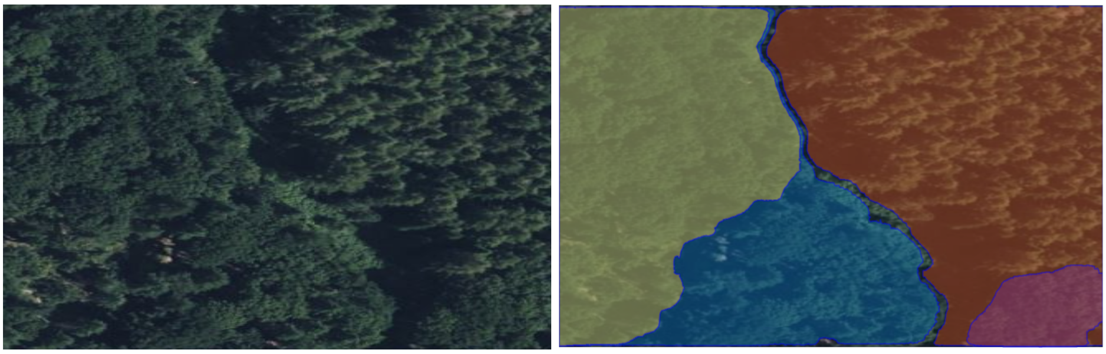
Other relevant indices, such as the Triangular Vegetation Index (TVI) and the Soil-Adjusted Vegetation Index (SAVI), were also considered. These indices help mitigate the effects of soil background and atmospheric conditions, further improving the accuracy of species classification and health assessment [2].

3.5 Data preparation for Object Detection

In addition to supervised learning models, YOLO (You Only Look Once) and Faster RCNN was used for object detection and species classification.

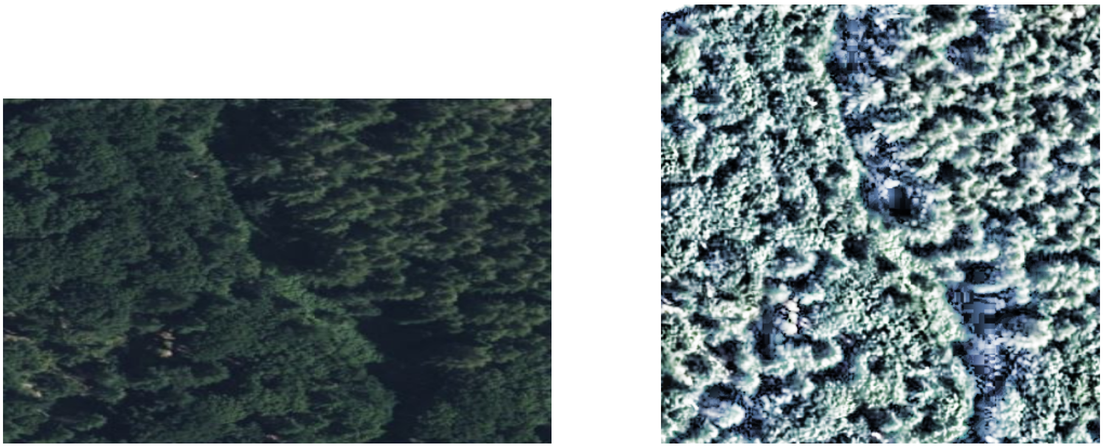
Annotation with Roboflow

The first step in applying YOLO, involved annotating images with bounding boxes around individual trees and labeling them with species names. The images used were random snapshots of the satellite map. It was also overlayed on multi-spectral 8-band imagery with 300dpi and 1:300 zoom resolution. This manual annotation was performed using Roboflow software [1], which facilitated the creation of a training dataset for the object detection models. Manual annotation was opted instead of using segementation models because, they did not yeild expected results. Fig 3.2b below shows one of the best segemented image by the-state-of-art SAM (Segment Anything Model) model [24], which misinterprets same species as different ones and the rest were even more fallacious. Given the time-intensive nature of manual annotation, a sampling strategy was employed to annotate 250 images with unique 22365 annotations of tree species, 6,583-CP, 6,269-NS, 2,161-OK, 4,887-DF, 1,297-BE, 1,168-BI, respectively.



(a) Original (b) Segmented

Figure 3.2: SAM output



(a) Original (b) AE enhanced and resized

Figure 3.3: AE output

Data Augmentation

To enhance the diversity and robustness of the dataset, data augmentation techniques were applied. These included transformations such as rotation, scaling, flipping, and color adjustments, namely Adaptive equalization (AE) , where it adjusts image contrast by analyzing local pixel values, enhancing areas with varying contrast to improve overall visibility and balance without losing detail (Fig 3.3b). Data augmentation helps to improve the generalization of the object detection model and ensures that it performs well under varying conditions [35].

Chapter 4

Methodology

4.1 Models

In this section, we will see in detail about the frameworks of the models used and how data flows through them, in our case.

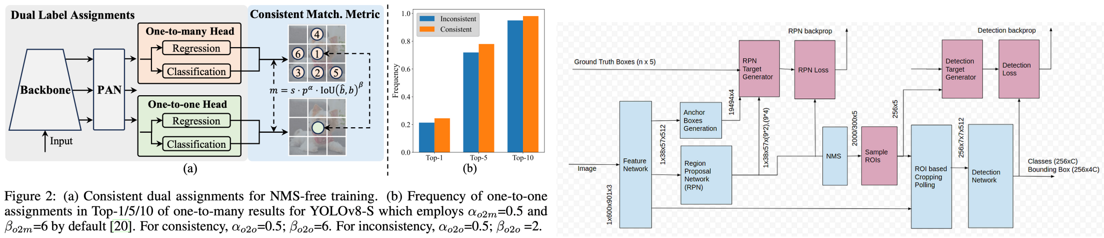
4.1.1 YOLOv10

YOLOv10 (Fig 4.1a) enhances its predecessors with several key innovations [39]. Its backbone features an advanced version of CSPNet (Cross Stage Partial Network), optimizing gradient flow and reducing computational redundancy. The neck component aggregates features from various scales using PAN (Path Aggregation Network) layers, which improves multiscale feature fusion and processing.

For predictions, YOLOv10 employs two types of heads. The One-to-Many Head generates multiple predictions per object during training, providing rich supervisory signals that enhance learning accuracy. During inference, the One-to-One Head produces a single, optimal prediction per object, eliminating the need for Non-Maximum Suppression (NMS). This approach not only reduces latency but also improves overall efficiency by streamlining the prediction process. These innovations collectively contribute to YOLOv10's superior performance and efficiency.

4.1.2 Faster RCNN

Faster R-CNN [16](Fig 4.1b) is a two-stage object detection model with a streamlined architecture. The model begins with a backbone network, often a pre-trained CNN such



(a) YOLOv10 architecture [22] (b) Faster RCNN architecture [16]

Figure 4.1: Object detection models architecture.

as ResNet, which performs feature extraction. This backbone generates feature maps that are essential for the subsequent stages of detection.

The second component is the Region Proposal Network (RPN), a fully convolutional network that operates on these feature maps. The RPN slides a small network over the feature maps, predicting multiple region proposals at each location, along with their ”objectness” scores, which indicate the likelihood of each region containing an object.

The final stage is the Fast R-CNN detector. It takes the proposed regions from the RPN and uses RoI (Region of Interest) pooling to extract fixed-size feature maps for each proposal. These feature maps are then processed through fully connected layers to predict the object’s class and refine the bounding box. By sharing convolutional features between the RPN and Fast R-CNN detector, Faster R-CNN achieves high efficiency in object detection.

4.1.3 Random Forest

Random Forest is an ensemble learning method that combines multiple decision trees to improve predictive performance. The architecture of Random Forest involves training several decision trees, each built on a random subset of the data and features. This process ensures that the model captures a diverse range of patterns and reduces the risk of overfitting.

A key aspect of Random Forest is Bootstrap Aggregating, or Bagging. Each decision tree is trained on a bootstrap sample, which is a random sample of the data drawn with replacement. This technique ensures that each tree is trained on slightly different data, contributing to the overall robustness of the model. In constructing these trees, Random Forest does not prune them, allowing each tree to grow to its maximum depth. For predictions, the model aggregates the outputs from all trees. In classification tasks, it uses the majority vote of all trees to determine the final class.

4.1.4 Neural networks

A basic feed-forward neural network is structured with an input layer, one or more hidden layers, and an output layer. The input layer receives the initial data, which is then passed through subsequent layers for processing. Each layer consists of nodes, or neurons, which are interconnected with nodes in adjacent layers. In the hidden layers, each neuron processes information by applying a weighted sum of its inputs, followed by an activation function. This process introduces non-linearity to the model, allowing it to learn complex patterns. The output layer generates the final prediction based on the processed information.

The network's performance is influenced by weights and biases, which are learnable parameters adjusted during training to minimize prediction errors. Activation functions, such as ReLU or sigmoid, are applied to the weighted sums at each neuron to introduce non-linearity.

4.1.5 XGBoost

XGBoost (Extreme Gradient Boosting) is a powerful implementation of gradient-boosted decision trees designed for regression tasks. Its architecture is based on the concept of sequentially adding weak learners, specifically decision trees, to improve the performance of the model by addressing the errors made by the existing trees.

The core components of XGBoost include decision trees as base learners, which are typically shallow to ensure flexibility and prevent overfitting. Gradient boosting is used to train each new tree to predict the residuals, or errors, of the current ensemble, effectively refining the model incrementally.

To prevent overfitting, XGBoost incorporates regularization with L1 and L2 terms in its objective function. Additionally, it calculates feature importance scores to assess and leverage the impact of each feature. The model also benefits from parallel and distributed computing, which accelerates the training process. Tree pruning is handled using a 'max_depth' parameter, allowing the model to prune trees backward to optimize performance. Furthermore, XGBoost includes a built-in method for handling missing values, enhancing its robustness and adaptability.

4.2 Evaluation methods

In this section we will discuss in brief, the evaluation metrics we will be using to understand our model's performance for object detection, pixel based detection and health prediction.

- **Precision (P):** Precision is the ratio of true positive detections to the total number of detections made (true positives + false positives). It measures the accuracy of the model in identifying relevant objects in images. The formula is:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}} \tag{4.1}$$

- **Recall (R):** Recall is the ratio of true positive detections to the total number of actual objects (true positives + false negatives). It evaluates the model's ability to detect all relevant objects in an image. The formula is:

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}} \tag{4.2}$$

- **Mean Average Precision (mAP):**mAP is the mean of the Average Precision (AP) scores across all classes in an image detection task. AP measures the area under the precision-recall curve for a specific class. mAP provides a comprehensive evaluation by considering both precision and recall across different detection thresholds. The formula is:

$$\text{AP} = \int_0^1 \text{Precision}(r)\,\mathrm{d}r \tag{4.3}$$

$$\text{mAP} = \frac{1}{N}\sum_{i=1}^{N} \text{AP}_i \tag{4.4}$$

  where N is the number of classes and AP_i is the average precision for the i-th class.

- **Accuracy:** Accuracy is the ratio of correctly predicted pixels (both true positives and true negatives) to the total number of pixels. It measures the overall correctness of the pixel classification in the image:

$$\text{Accuracy} = \frac{\text{True Positives (TP)} + \text{True Negatives (TN)}}{\text{Total Pixels}} \tag{4.5}$$

- **Mean Squared Error (MSE):** MSE measures the average squared difference between the predicted and actual values. It penalizes larger errors more severely, making it sensitive to outliers:

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(\text{Predicted}_i - \text{Actual}_i)^2 \tag{4.6}$$

  where N is the number of observations.

- **Mean Absolute Error (MAE):** MAE is the average of the absolute differences between predicted and actual values. Unlike MSE, it treats all errors equally and is less sensitive to outliers:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|\text{Predicted}_i - \text{Actual}_i| \tag{4.7}$$

- **R-squared (R²):** R² represents the proportion of variance in the dependent variable that is predictable from the independent variables. It provides an indication of the goodness of fit of the model, with values closer to 1 indicating better performance:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(\text{Actual}_i - \text{Predicted}_i)^2}{\sum_{i=1}^{N}(\text{Actual}_i - \text{Ac}\bar{\text{t}}\text{ual})^2} \tag{4.8}$$

  where Ac$\bar{\text{t}}$ual is the mean of the actual values.

Chapter 5

Experiment and Results

5.1 Experiment Environment

In this project, the classification of tree species and the detection of tree health utilized several models, including Random Forest (RF), XGBoost, and Neural Networks (NNs), as well as models for object detection. These models were implemented using Python, leveraging libraries such as Scikit-learn for RF and XGBoost, and TensorFlow 2.17 with its Keras library for the neural network approaches. The YOLO object detection model was applied using the Roboflow platform for annotation and dataset preparation. All experiments were conducted on an NVIDIA GeForce RTX 2060 GPU with CUDA 11.4, which facilitated efficient processing and training of the models.

5.2 Experiment Setting

This section details the experimental setup used to evaluate and compare different object detection models, including versions of the YOLO (You Only Look Once) model and Faster R-CNN. The focus was on assessing model performance across various scenarios, such as learning rate variations, data augmentation, model size differences, and training duration. All experiments were conducted using the AdamW optimizer with 200 epochs, incorporating early stopping with a patience of 50 epochs to avoid overfitting. Default settings were maintained for other parameters. And for pixel-based classification, Random forest classifer and tri-layer Neural network was tranined on datasets with and without vegetation indices namely, NDVI, TVI, SAVI.

We first compared the YOLOv10 model to its predecessor, YOLOv8, to assess improvements in accuracy and efficiency. Key metrics like mean Average Precision (mAP), precision, recall, and inference time were used to evaluate both models over 200 epochs. The comparison, was later visualized through performance curves.

Next we wanted to examine the impact of model size on YOLOv10's performance by comparing its small and large variants, revealing the trade-offs between performance and resource consumption. The results were visualized with graphs to show how model size influences YOLOv10's effectiveness.

Another important hyperparameter to be fine-tuned is learning rate, so we tested the impact of different learning rates (1e-3, 1e-4, and 1e-5) on YOLOv10 and Faster R-CNN, focusing on convergence speed, final accuracy, and stability. Both models were trained with the AdamW optimizer for 200 epochs. The results, later were summarized, to identify the optimal learning rate for each model, aiding in fine-tuning for better performance.

Increase in the number of epochs trained, a model's performance may vary positive or negative. To assess the impact of prolonged training, both YOLOv10 and Faster R-CNN were trained for up to 200 epochs with early stopping disabled. This experiment provided insights into convergence behavior, final performance, and potential overfitting.

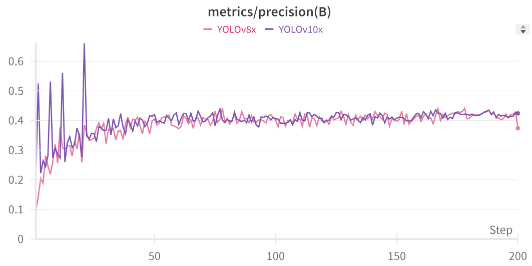
The fifth experiment evaluated the effects of data augmentation on YOLOv10 and Faster R-CNN. Techniques like rotation, flipping, and color adjustments (AE) were applied to the dataset. Performance was compared between original (250 images) and augmented (954 images) datasets, with the impact visualized through performance curves, showing changes in mAP, precision, and recall.

The final experiment involved a species-wise analysis of the best-performing model from the previous tests. This model was evaluated on its ability to correctly identify and classify different species. An image was generated to visualize the model's species-wise detection capabilities, offering insights into its practical application for specific detection tasks.

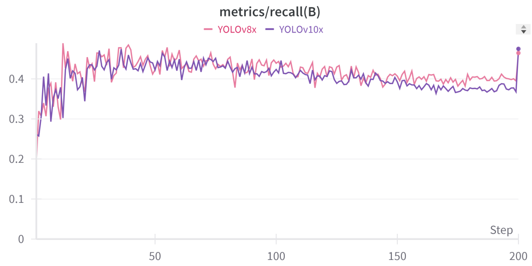
These experiments provided valuable insights into the performance of YOLO and Faster R-CNN models, guiding improvements in object detection models.

For pixel-based classification, the datasets used in these experiments were divided into two categories: one with vegetation indexes (such as NDVI, TVI, etc.) and one without them. Vegetation indexes are often used in remote sensing and environmental monitoring to quantify vegetation cover, health, and other characteristics. And the RF model was used with default configuration, whereas the 3-layered NN model was used without adding regularizations, as it was tested and model did not converge with them.

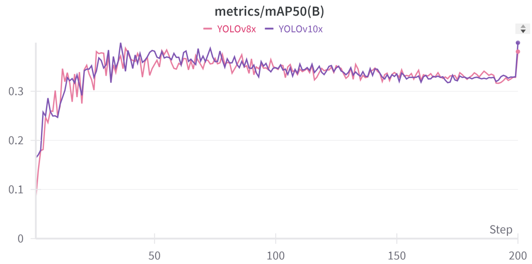
And finally for health prediction, the XGBoost model was tested on data which



(a) Precision



(b) Recall



(c) mAP

Figure 5.1: YOLOv8 vs YOLOv10

Models	Precision	Recall	mAP
YOLOv8	0.374	0.465	0.381
YOLOv10	**0.424**	**0.475**	**0.399**

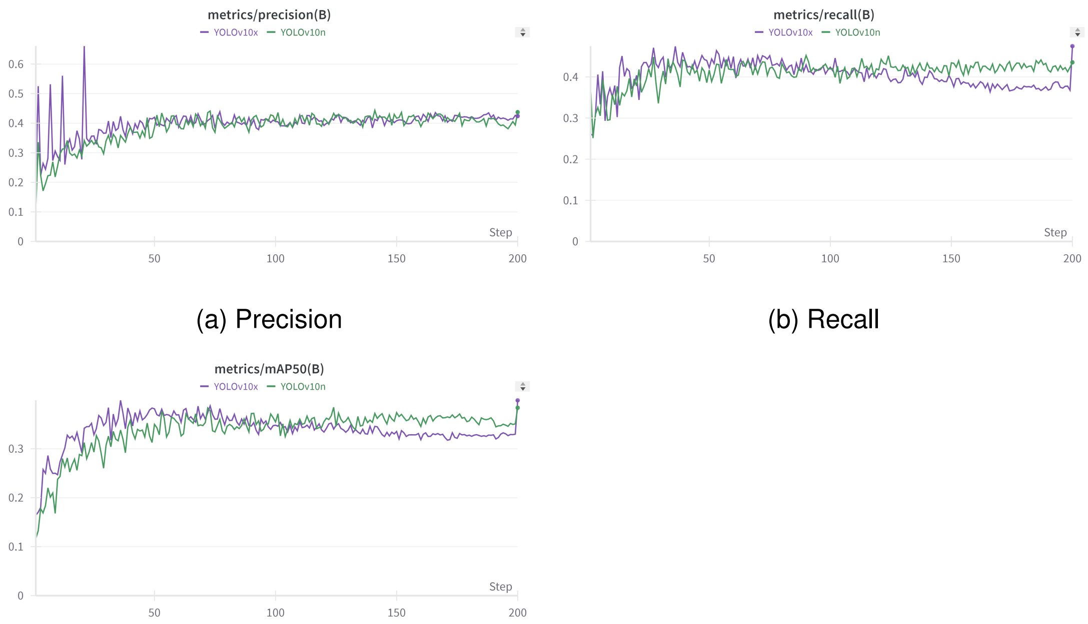
Table 5.1: Models comparison. Both models were trained on original data with default configuration.

included LAI values and ran on default configuration, with n_estimators alone set to 400.

5.3 Results

5.3.1 Results for different YOLO versions

In this experiment, different YOLO versions namely v8 and v10 (latest YOLO version) was tested with default parameter configuration on non-augumented original data.



(a) Precision

(b) Recall

(c) mAP

Figure 5.2: YOLOv10x vs YOLOv10n

Models	Precision	Recall	mAP
YOLOv10n	0.437	0.435	0.384
YOLOv10x	**0.424**	**0.475**	**0.399**

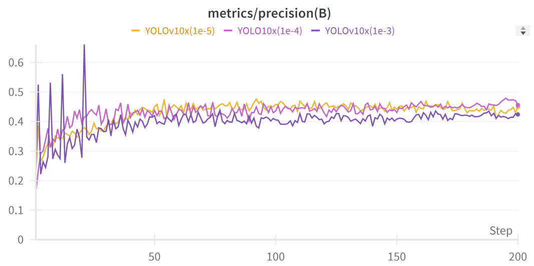
Table 5.2: Models comparison. Both models were trained on original data with default configuration.

5.3.2 Results for different YOLOv10 model sizes

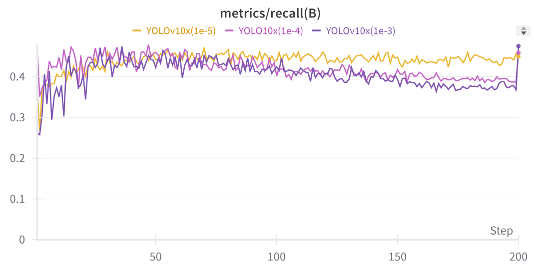
In this experiment, we compare YOLO version 10-N, which is a smaller version with 2.3M params and version 10-X, which is the largest version with 29.5M params.

5.3.3 Results for different learning rates

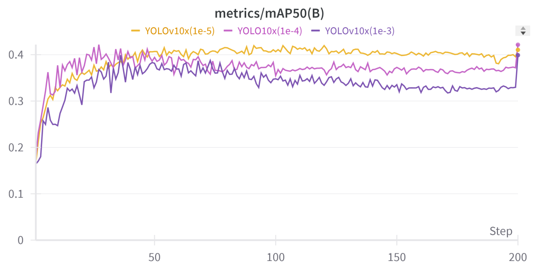
In this experiment, YOLOv10x and Faster-RCNN was trained using AdamW optimizer with various learning rates (1e-3, 1e-4, 1e-5), for 200 epochs and 3000 iterations, respectively. Both the models were trained on data with early-stopping disbled for this experiment.



(a) Precision



(b) Recall

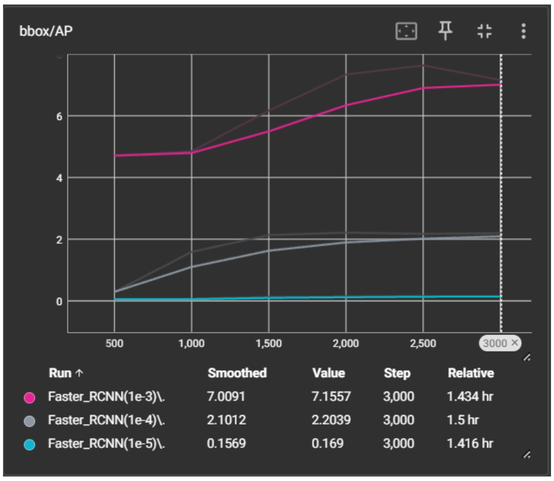


(c) mAP

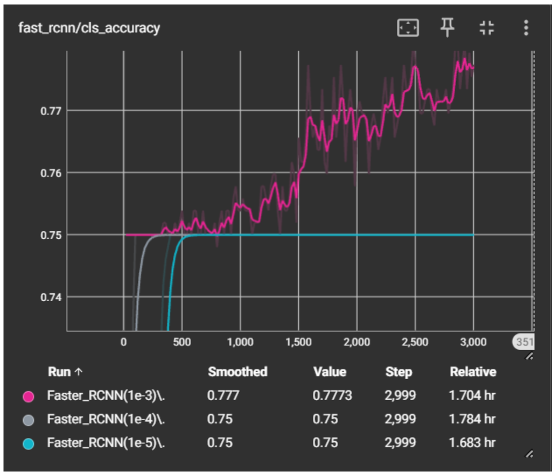
Figure 5.3: AdamW optimizer with different learning rates (1e-3, 1e-4, 1e-5) tested on YOLOv10x. It is trained on data for 200 epochs.

Models	Learning rate	Precision	Recall	mAP	Class_Accuracy
	1e-3	42.4%	47.5%	39.9%	-
YOLOv10x	1e-4	45.0%	44.9%	41.0%	-
	1e-5	**45.6%**	**45.9%**	**42.2%**	**-**
	1e-3	7.156%	-	19.956%	0.777
Faster-RCNN	1e-4	2.101%	-	6.6378%	0.75
	1e-5	0.169%	-	0.5828%	0.75

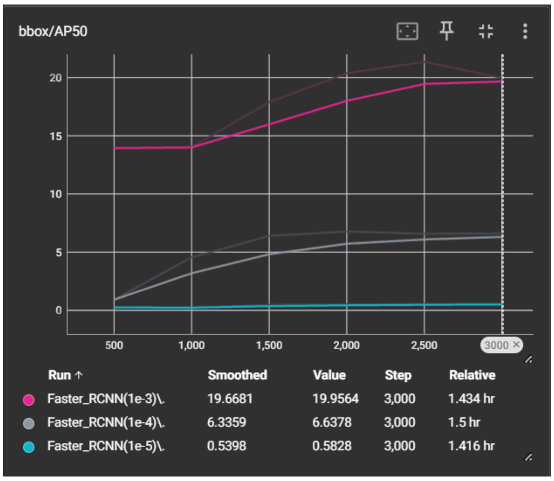
Table 5.3: Table shows YOLOv10x and Faster-RCNN models trained on data with AdamW optimizer using different learning rates (1e-3, 1e-4, 1e-5) for 200 epochs and 3000 iteratons respectively. Early stopping was disabled for this test.



(a) Precision

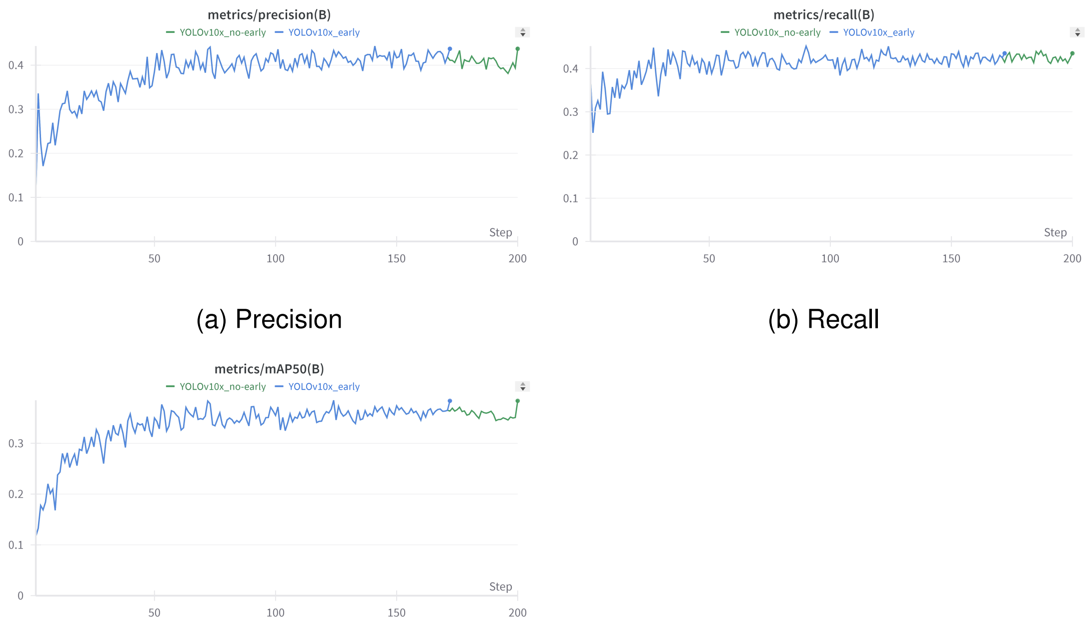


(b) Class_Accuracy



(c) mAP

Figure 5.4: AdamW optimizer with different learning rates (1e-3, 1e-4, 1e-5) tested on Faster-RCNN. It is trained on data for 3000 iterations. This model alone was visualized using TensorBoard as it did not support WandB.



(a) Precision

(b) Recall

(c) mAP

Figure 5.5: Best model was trained on data for 200 epochs each with early stopping enabled and disabled

Models	Patience	Precision	Recall	mAP
YOLOv10x-early	50	0.456	0.459	0.422
YOLOv10x-no-early	0	0.456	0.459	0.422

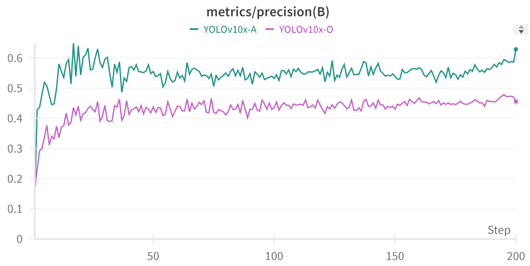
Table 5.4: Table shows best model (YOLOv10x with 1e-5 learning rate) performance on data for 200 epochs each, with early stopping enabled and disabled.

5.3.4 Results for model on larger epochs

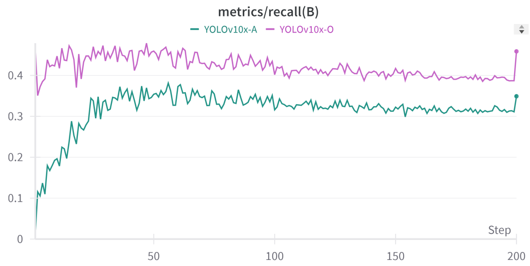
This experiment was run on model with and without early-stopping enabled, for 200 epochs on data.

5.3.5 Results of Best model on Original and Augmented data

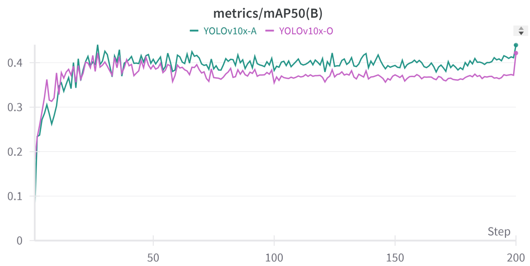
In this experiment, the best model (YOLOv10x with learning rate 1e-5) from previous experiments was selected and trained on original and augmented data to see it's performance on both the datasets.



(a) Precision



(b) Recall

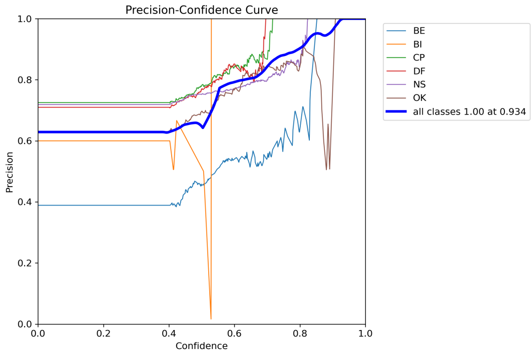


(c) mAP

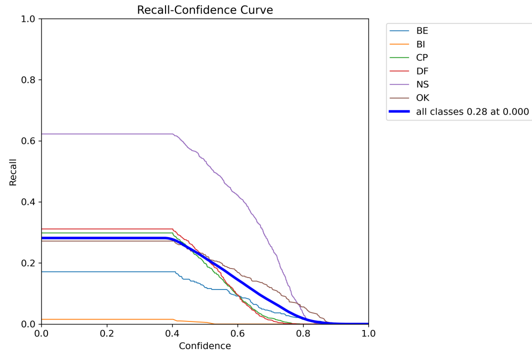
Figure 5.6: Best model (YOLOv10x with 1e-5 learning rate) was trained on original data and augmented data for 200 epochs each with early stopping disabled.

Models	Precision	Recall	mAP
YOLOv10x-O	0.456	0.459	0.422
YOLOv10x-A	**0.629**	**0.350**	**0.440**

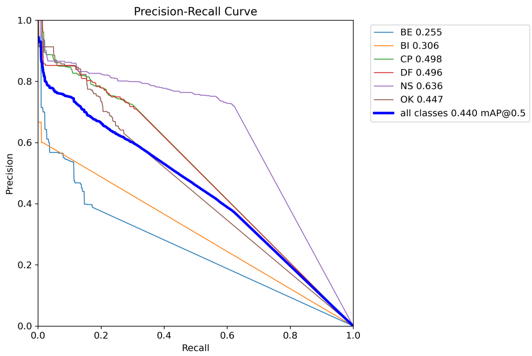
Table 5.5: Table shows best model (YOLOv10x with 1e-5 learning rate) performance on original data and augmented data for 200 epochs each, with early stopping disabled.



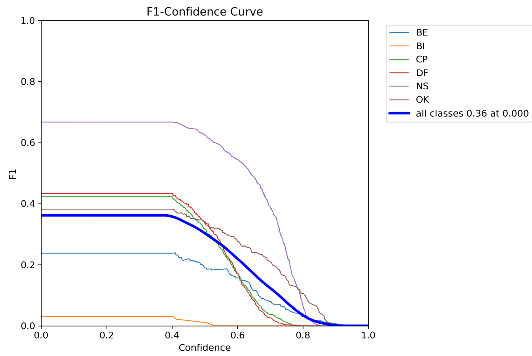
(a) Precision vs Confidence



(b) Recall vs Confidence



(c) Precision vs Recall



(d) F1-score vs Confidence

Figure 5.7: Species-wise performance of the best model was visualised, showing Precision, recall and F1-score against confidence scores.BI, BE, OK have relatively lower metric curves due to their low instance counts in training data.

5.3.6 Results for species-wise analysis

In this experiment, species-wise results were tested on best model, to understand the model in-depth. So, precision, recall and F1 curves were drawn against confidence curves.

5.3.7 Results with and without Vegetation indices

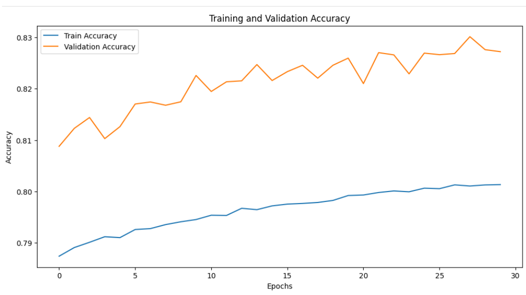
In this experiment Random forest and 3-layered Neural network was trained with and without vegetation indices.

5.3.8 Results for tree health prediction

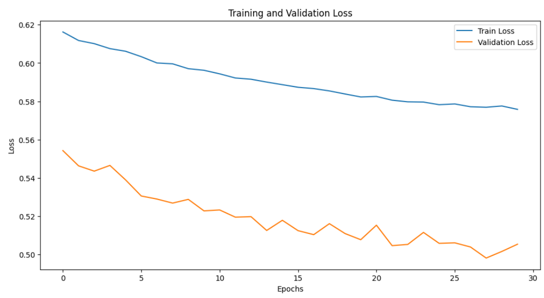
In this experiment, dataset with LAI values are tested on XGBoost model.

Class	Precision	Recall	mAP
BE	0.39	0.18	0.26
BI	0.60	0.05	0.31
CP	0.73	0.30	0.50
DF	0.71	0.31	0.50
NS	0.72	0.63	0.64
OK	0.63	0.28	0.45

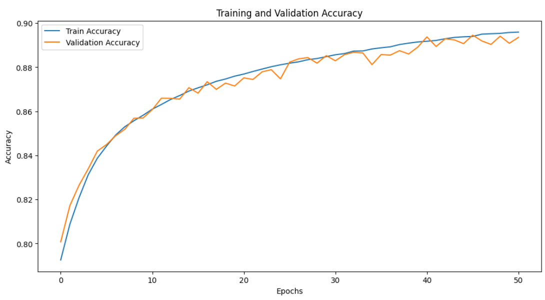
Table 5.6: Table shows best model (YOLOv10x with 1e-5 learning rate) performance class-wise.



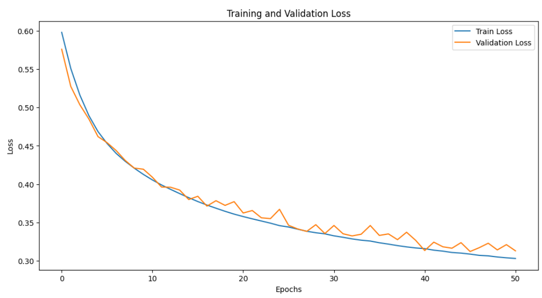
(a) Train_accuracy vs Val_accuracy-NN



(b) Train_loss vs Val_loss-NN



(c) Train_accuracy vs Val_accuracy-NN_VI

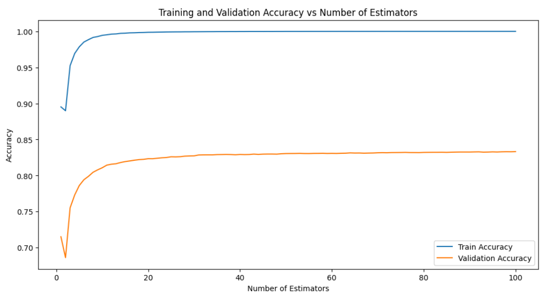


(d) Train_loss vs Val_loss-NN_VI

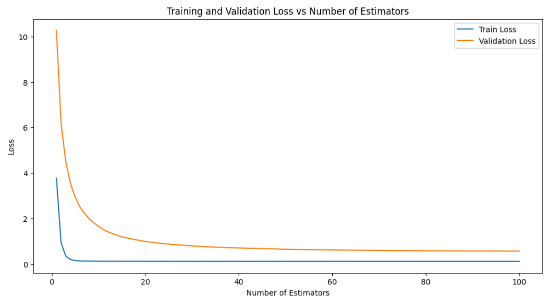
Figure 5.8: 3-layer neural network, with and without vegetation indices.

Model	Class	Precision	Recall	F1-score	Model	Class	Precision	Recall	F1-score
	BE	0.81	0.42	0.55		BE	0.86	0.47	0.61
	BI	0.75	0.52	0.61		BI	0.79	0.57	0.66
RF	CP	0.78	0.71	0.75	RF_VI	CP	0.83	0.77	0.80
	DF	0.87	0.88	0.88		DF	0.88	0.89	0.89
	NS	0.85	0.83	0.84		NS	0.87	0.84	0.85
	OK	0.83	0.94	0.88		OK	0.84	0.95	0.89
ACCURACY			0.83		ACCURACY			0.85	
	BE	0.84	0.95	0.89		BE	0.83	0.66	0.74
	BI	0.73	0.43	0.54		BI	0.81	0.71	0.75
NN	CP	0.74	0.68	0.71	**NN_VI**	CP	0.86	0.83	0.84
	DF	0.86	0.88	0.87		DF	0.93	0.92	0.93
	NS	0.81	0.85	0.83		NS	0.89	0.91	0.90
	OK	0.85	0.93	0.89		OK	0.90	0.95	0.93
ACCURACY			083		ACCURACY			**0.89**	

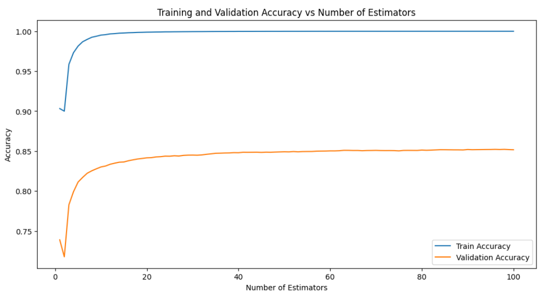
Table 5.7: Table shows pixel-based model performance with and without vegetation indices(VI).



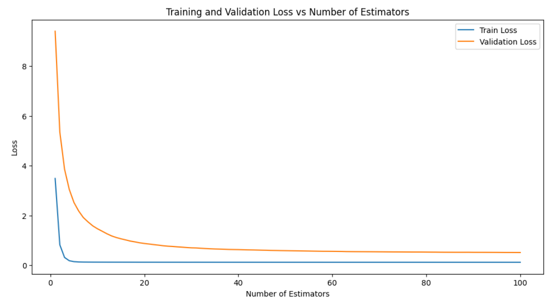
(a) Train_accuracy vs Val_accuracy-RF



(b) Train_loss vs Val_loss-RF



(c) Train_accuracy vs Val_accuracy-RF_VI



(d) Train_loss vs Val_loss-RF_VI

Figure 5.9: Random forest, with and without vegetation indices.

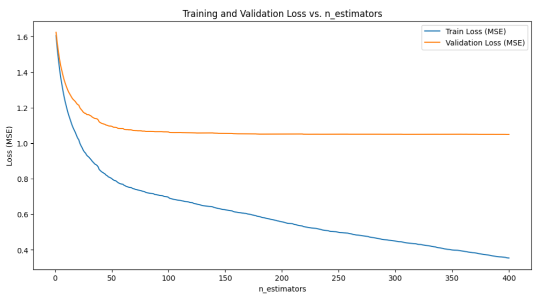


Figure 5.10: MSE vs n_estimators, for XG-Boost model.

MSE	MAE	R^2
1.0497	0.7575	0.6804

Table 5.8: XGBoost performance

Chapter 6

Discussion and Analysis

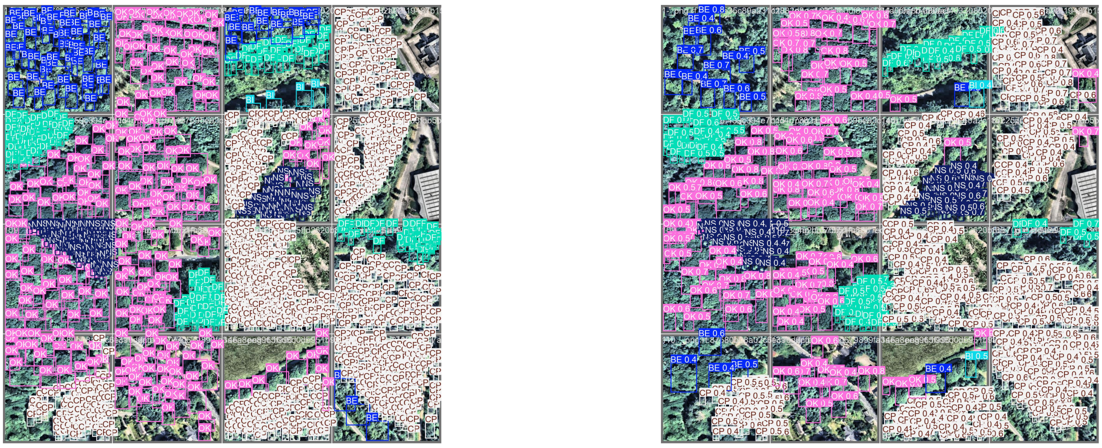
From Tab 5.1, YOLOv10 architecture illustrated it is a slightly superior model to YOLOv8 in regards to all the metrics, which can also be seen in Fig 5.1a,5.1b,5.1c. From Tab 5.2, YOLOv10n seems to be slightly better at avoiding false positives, making it a model with higher precision [0.437]. Fig 5.2a shows, the model avoids incorrect detections. YOLOv10x, on the other hand, is more successful at detecting a higher percentage of actual objects (higher recall [0.475]) (Fig 5.2b) and has a better overall performance (higher mAP [0.399])(Fig 5.2c). This makes it more suitable for applications where it is more important to ensure that most objects are detected, even at the cost of a few more false positives.

From Tab 5.3, in Yolov10x, a relatively higher learning rate of 1e-3 yields moderate performance. The model achieves decent recall but slightly lower precision [42.4%] and mAP [39.9%]. This suggests that while the model is good at detecting objects, it may be prone to overfitting or instability in the learning process, leading to slightly lower overall accuracy.A learning rate of 1e-4 improves precision and mAP compared to 1e-3, indicating a more stable learning process. The model achieves a better balance between precision and recall, leading to improved overall performance.The lowest learning rate (1e-5) provides the best performance among the three. Both precision [45.6%] and recall [45.9%] are maximized, leading to the highest mAP [42.2%]. This suggests that the model benefits from a slower, more controlled learning process, allowing for better convergence and fine-tuning of weights, shown in Fig 5.3a,5.3b,5.3c. Whereas in Faster RCNN, shown in Fig 5.4a,5.5b,5.4c, the model struggles with a higher learning rate, showing very low precision and mAP, though class accuracy is somewhat reasonable. Class accuracy here denotes the class assigned to the detected trees. This suggests that the model might be unstable and unable to converge properly. Lowering the learning rate to 1e-4 results in a significant drop in both precision and mAP, indicating that the model's performance is deteriorating further. The class accuracy remains stable, which suggests that the model is struggling with detecting objects but still somewhat correctly classifying them. At the lowest learning rate, the model performs the worst, with near-zero precision and mAP. This indicates that the learning rate is too low for the model to make meaningful updates to its parameters, leading to poor performance overall. As to summarise, YOLOv10x is much more resilient to changes in learning rate and achieves better performance overall compared to Faster-RCNN. It benefits from a lower learning rate (1e-5), which allows it to fine-tune and improve precision, recall, and mAP effectively. Faster-RCNN, however, does not perform well with the AdamW optimizer at any of the tested learning rates, struggling particularly with lower learning rates. This might indicate that Faster-RCNN requires a different approach in terms of optimization or that it is less suited to this specific task or dataset compared to YOLOv10x.

The results from the Tab 5.4 suggest that, for the YOLOv10x model trained on augmented data with a learning rate of 1e-5, early stopping (Fig 5.6c) did not provide any additional benefit in terms of performance metrics. Both models performed equally well, indicating that the model's training was stable and did not suffer from overfitting or unnecessary prolongation of training. This could imply that, under similar circumstances, early stopping may be optional rather than necessary, particularly when using effective regularization techniques like data augmentation. However, early stopping might still be valuable for saving computational resources and time, especially when working with larger datasets or longer training cycles.

In the next experiment, when the model was trained on the augmented dataset (Tab 5.5), all performance metrics improved except for recall, shown in Fig 5.6a,5.6b,5.6c. The precision increased to 0.629, mAP to 0.440, and recall decreased to 0.350. The improvement in precision suggests that the model became slightly better at correctly identifying true positives, with fewer incorrect predictions. The decrease in recall indicates that the model was not able to detect a greater number of objects in the dataset, this behaviour can be attributed to the increase of model's precision with increased data. The higher mAP further confirms that the model's overall ability to detect and localize objects improved as a result of training on the augmented data.

In pixel-based classification, the results (Tab 5.7 and Fig 5.8,5.9), clearly demonstrate that incorporating Vegetation Indices into the training data can significantly enhance the performance of models, with NN_VI achieving the highest at 89% accuracy



(a) True labels (b) Predicted labels

Figure 6.1: YOLOv10x (1e-5). Original labels on left and predictions on right.

(Fig 5.8c). VI likely provided additional spectral information that is highly relevant to vegetation, allowing the models to better distinguish between different species based on their unique spectral signatures. This is particularly beneficial in this ecosystem where different species may have similar visual characteristics but different spectral profiles, the reason why, object detection model struggles and these models outstrips them in performance. And the reason, oak (OK) has best metrics among all classes is because, it has the most unique values in data, but when image snapshots were taken from the satellite, norway spruce (NS) had more recurring instances, making it better in object detection models.

In health prediction, Leaf area index(LAI) value between 3-6 is considered healthy, or otherwise. The regression model (Tab 5.8) demonstrates a solid fit to the data, with a high R^2 value indicating that it explains a substantial portion of the variance in the target variable (LAI). The error metrics (MSE and MAE) are relatively low (Fig 5.10), pointing to reasonably accurate predictions.

Understanding the Model predictions and limitations

From Figure 6.1b, we can see the YOLOv10x model predicts fewer tree objects in the image (about 81 instances on average in test images, but only about 35-50 instances on average in predicted images) but correctly identifies most of those it does detect (see Fig 5.6a,5.6b), this behavior can be attributed to the interplay between the model's confidence threshold and Intersection Over Union (IoU) threshold. The confidence threshold in YOLOv10x determines the minimum probability that a predicted bounding box must have for the model to consider it a valid detection. If this threshold is set too high, 0.6 (default) in our case, the model only output predictions it is very certain about, potentially leading to fewer detections overall. This behavior can explain why the model predicts fewer tree objects, it is likely discarding detections that fall below the high confidence threshold, even if they are correct. On the other hand, IoU threshold is used during the Non-Maximum Suppression (NMS) process to eliminate redundant or overlapping bounding boxes. If the IoU threshold is set too high, 0.6 (default), the model discards additional bounding boxes that have significant overlap with others, even if they correspond to different tree objects. This can result in fewer detections, particularly in our case where trees are closely spaced or overlapping.

Detecting trees in heterogeneous and dense forests is inherently challenging, and these difficulties are amplified in our case, where both factors are combined. The complexity of such environments arises from the wide variety of tree shapes, sizes, appearances, overlapping canopies, diverse species, and different growth stages. This variability complicates the task for object detection models, which may struggle to consistently identify and differentiate between individual trees, leading to errors such as missed detections or incorrect classifications. The dense foliage and shadows further obscure trees, exacerbating the difficulty of accurate detection. When using data directly from raw satellite imagery (even when overlayed over SuperDove imagery) instead of UAVs or drones, these challenges are exacerbated. Satellite images typically have lower spatial resolution compared to images captured by UAVs, making it harder to discern individual trees, especially in dense forests. The large-scale perspective of satellite imagery often results in pixelated or blurry representations of trees, reducing the model's ability to detect fine details. Moreover, atmospheric conditions, lighting variations, tree-crown angle variations and the presence of clouds can further degrade the quality of satellite images, increasing the difficulty for object detection models to accurately identify trees in such complex and cluttered environments.

But still, this data can be valuable for training models that are robust to the specific issues encountered in satellite imagery, such as low resolution, high variability, and atmospheric distortions. It allows for the development of techniques that enhance model performance and multi-scale feature extraction. Furthermore, satellite imagery datasets can be used to create synthetic or semi-synthetic training sets by combining real satellite data with generated annotations, which can help in training models to better generalize. By utilizing such data as a benchmark or training resource, more accurate and reliable object detection for complex and challenging environments are made.

Chapter 7

Conclusion

In this study, we explored various models for detecting tree species, including Random Forest (RF), Neural Networks, and Faster R-CNN, while also innovatively applying the latest YOLOv10. The YOLOv10x model particularly showcased a balance of strengths and limitations within the challenging environment of dense, heterogeneous forests. It achieved high precision and recall, but its tendency to predict fewer tree objects was influenced by the confidence threshold and Intersection Over Union (IoU) threshold. A high confidence threshold limited the model to only the most certain detections, potentially overlooking valid trees, while a high IoU threshold excluded overlapping bounding boxes, critical in our dense forest scenario. Despite predicting fewer instances, YOLOv10x maintained impressive accuracy in its detections.

In pixel-based classification, the integration of Vegetation Indices (VI) significantly enhanced performance. VI provided additional spectral data, improving the model's ability to differentiate between tree species, especially where visual characteristics were insufficient. The NN_VI model, leveraging this spectral information, achieved the highest accuracy of 89%, demonstrating VI's effectiveness in refining classification.

For health prediction, the regression model performed well, with a high R^2 value indicating strong explanatory power over the Leaf Area Index (LAI). Low MSE and MAE further validated the model's accuracy.

While the NN_VI model excelled in classification with 89% accuracy (greater than models that are used in the Edinburgh forest department with 83% accuracy and close to other SOTA models that were trained on public inventories and global datasets), YOLOv10x stood out, showing exceptional adaptability and robustness, despite the parameter and data complexities. It may very well outperform other models, with sufficient high quality data and resources to optimize IOU and confidence threshold.

Chapter 8

Future work

Given the complexities encountered in this study, avenues for future research are evident. First, the data itself presents opportunities for improvement. While satellite imagery offers a broad overview, its limitations in resolution and clarity suggest that combining it with higher-resolution UAV or drone data could yield better results. Developing techniques for data fusion, where satellite and UAV data are combined to produce richer datasets, could significantly enhance model performance.

If YOLOv10 is used, fine-tuning the confidence and IoU thresholds more dynamically during training or developing adaptive mechanisms that adjust these parameters based on the density and overlap of objects is crucial. Additionally, experimenting with multi-scale feature extraction techniques might improve the model's ability to detect smaller or partially obscured trees, which are often missed in current approaches.

Next, the application of ensemble learning techniques, combining models like YOLOv10x with VI-enhanced classifiers, could be explored to leverage the strengths of both approaches, potentially leading to even higher accuracy and robustness.

Finally, finding ways to leverage SOTA segmentation algorithms or modelling segmentation algorithms, is crucial in regards to annotation, as it would save the effort of manual annotation. Moreover, it would simply be impossible to manually annotate, if the images in the dataset exceed a certain threshold.

Bibliography

[1] Dataset,roboflow.

[2] Remote sensing indices.

[3] Hervé Abdi, Dominique Valentin, and Betty Edelman. *Neural networks*. Number 124. Sage, 1999.

[4] Oludare Isaac Abiodun, Aman Jantan, Abiodun Esther Omolara, Kemi Victoria Dada, Nachaat AbdElatif Mohamed, and Humaira Arshad. State-of-the-art in artificial neural network applications: A survey. *Heliyon*, 4(11), 2018.

[5] Siham Acharki. Planetscope contributions compared to sentinel-2, and landsat-8 for lulc mapping. *Remote Sensing Applications: Society and Environment*, 27:100774, 2022.

[6] Abdulaziz Amer Aleissaee, Amandeep Kumar, Rao Muhammad Anwer, Salman Khan, Hisham Cholakkal, Gui-Song Xia, and Fahad Shahbaz Khan. Transformers in remote sensing: A survey. *Remote Sensing*, 15(7):1860, 2023.

[7] Kai O. Bergmüller and Mark C. Vanderwel. Predicting tree mortality using spectral indices derived from multispectral uav imagery. *Remote Sensing*, 14(9), 2022.

[8] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[9] Jick Castanha, Indrawati, Subhash K.B. Pillai, Gadang Ramantoko, and Tri Widarmanti. A systematic literature review on natural language processing (nlp). In *2022 International Conference on Advanced Creative Networks and Intelligent Systems (ICACNIS)*, pages 1–6, 2022.

[10] Sungeun Cha, Joongbin Lim, Kyoung-Min Kim, Jongsoo Yim, and Woo kyun Lee. Uncovering the potential of multi-temporally integrated satellite imagery for accurate tree species classification. *Forests*, 2023.

[11] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[12] Kwanghun Choi, Wontaek Lim, Byungwoo Chang, Jinah Jeong, Inyoo Kim, Chan-Ryul Park, and Dongwook W Ko. An automatic approach for tree species detection and profile estimation of urban street trees using deep learning and google street view images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 190:165–180, 2022.

[13] Matthew L Clark and Dar A Roberts. Species-level differences in hyperspectral metrics among tropical rainforest trees as determined by a tree-based classifier. *Remote Sensing*, 4(6):1820–1855, 2012.

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[15] Matthias Drusch, Umberto Del Bello, Sébastien Carlier, Olivier Colin, Veronica Fernandez, Ferran Gascon, Bianca Hoersch, Claudia Isola, Paolo Laberinti, Philippe Martimort, et al. Sentinel-2: Esa's optical high-resolution mission for gmes operational services. *Remote sensing of Environment*, 120:25–36, 2012.

[16] Ross Girshick, Ilija Radosavovic, Georgia Gkioxari, Piotr Dollár, and Kaiming He. Detectron. https://github.com/facebookresearch/detectron, 2018.

[17] Pall Oskar Gislason, Jon Atli Benediktsson, and Johannes R Sveinsson. Random forests for land cover classification. *Pattern recognition letters*, 27(4):294–300, 2006.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[19] John Hogland, Nathaniel Anderson, David L. R. Affleck, and Joseph St. Peter. Using forest inventory data with landsat 8 imagery to map longleaf pine forest characteristics in georgia, usa. *Remote. Sens.*, 11:1803, 2019.

[20] Linsheng Huang, Yong Liu, Wenjiang Huang, Yingying Dong, Huiqin Ma, Kang Wu, and Anting Guo. Combining random forest and xgboost methods in detecting early and mid-term winter wheat stripe rust using canopy level hyperspectral measurements. *Agriculture*, 12(1):74, 2022.

[21] Sha Huang, Lina Tang, Joseph P Hupy, Yang Wang, and Guofan Shao. A commentary review on the use of normalized difference vegetation index (ndvi) in the era of popular remote sensing. *Journal of Forestry Research*, 32(1):1–6, 2021.

[22] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics YOLO, January 2023.

[23] Aydin Kaya, Ali Seydi Keceli, Cagatay Catal, Hamdi Yalin Yalic, Huseyin Temucin, and Bedir Tekinerdogan. Analysis of transfer learning for deep neural network based plant classification models. *Computers and Electronics in Agriculture*, 158:20–29, 2019.

[24] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023.

[25] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

[26] GISP Kurt Menke, Richard Smith Jr, Luigi Pirelli, GISP John Van Hoesen, et al. *Mastering QGIS*. Packt Publishing Ltd, 2016.

[27] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[28] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 eighth ieee international conference on data mining*, pages 413–422. IEEE, 2008.

[29] Xiaoye Liu, Zhenyu Zhang, Jim Peterson, and Shobhit Chandra. The effect of lidar data density on dem accuracy. In *Proceedings of the 17th International Congress on Modelling and Simulation (MODSIM07)*. Modelling and Simulation Society of Australia and New Zealand, 2007.

[30] H Łoś, G Sousa Mendes, David Cordeiro, Nuno Grosso, Hugo Costa, Pedro Benevides, and Mário Caetano. Evaluation of xgboost and lgbm performance in tree species classification with sentinel-2 data. In *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*, pages 5803–5806. IEEE, 2021.

[31] Janne Mäyrä, Sarita Keski-Saari, Sonja Kivinen, Topi Tanhuanpää, Pekka Hurskainen, Peter Kullberg, Laura Poikolainen, Arto Viinikka, Sakari Tuominen, Timo Kumpula, and Petteri Vihervaara. Tree species classification from airborne hyperspectral and lidar data using 3d convolutional neural networks. *Remote Sensing of Environment*, 256:112322, 2021.

[32] John Nisbet. The history of the forest of dean, in gloucestershire. *The English Historical Review*, 21(83):445–459, 1906.

[33] Roman Novak, Yasaman Bahri, Daniel A Abolafia, Jeffrey Pennington, and Jascha Sohl-Dickstein. Sensitivity and generalization in neural networks: an empirical study. *arXiv preprint arXiv:1802.08760*, 2018.

[34] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[35] Stephen M Pizer, E Philip Amburn, John D Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart ter Haar Romeny, John B Zimmerman, and Karel Zuiderveld. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, 39(3):355–368, 1987.

[36] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[37] Renata Lopes Rosa, Gisele Maria Schwartz, Wilson Vicente Ruggiero, and Demóstenes Zegarra Rodríguez. A knowledge-based recommendation system that includes sentiment analysis and deep learning. *IEEE Transactions on Industrial Informatics*, 15:2124–2135, 2019.

[38] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.

[39] Ranjan Sapkota, Rizwan Qureshi, Marco Flores Calero, Chetan Badjugar, Upesh Nepal, Alwin Poulose, Peter Zeno, Uday Bhanu Prakash Vaddevolu, Sheheryar Khan, Maged Shoman, Hong Yan, and Manoj Karkee. Yolov10 to its genesis: A decadal and comprehensive review of the you only look once (yolo) series, 2024.

[40] Alana G Semple, Bin Tan, and Guoqing Lin. Geometric assessment of planetscope imagery. In *23rd Annual American Geophysical Union (AGU) MEeting*, 2023.

[41] Yun Shi, Donghui Ma, Jie Lv, and Jie Li. Actl: Asymmetric convolutional transfer learning for tree species identification based on deep neural network. *IEEE Access*, 9:13643–13654, 2021.

[42] Yu-Hsuan Tu, Kasper Johansen, Bruno Aragon, Marcel M El Hajj, and Matthew F McCabe. The radiometric accuracy of the 8-band multi-spectral surface reflectance from the planet superdove constellation. *International Journal of Applied Earth Observation and Geoinformation*, 114:103035, 2022.

[43] Blanka Vajsová, Dominique Fasbender, Csaba Wirnhardt, Slavko Lemajic, and Wim Devos. Assessing spatial limits of sentinel-2 data on arable crops in the context of checks by monitoring. *Remote Sensing*, 12(14):2195, 2020.

[44] Ashish Vaswani. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[45] Luisa Velasquez-Camacho, Maddi Etxegarai, and Sergio de Miguel. Implementing deep learning algorithms for urban tree detection and geolocation with high-resolution aerial, satellite, and ground-level images. *Computers, Environment and Urban Systems*, 105:102025, 2023.

[46] Athanasios Voulodimos, Nikolaos D. Doulamis, Anastasios D. Doulamis, and Eftychios E. Protopapadakis. Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, 2018, 2018.

[47] Ao Wang, Hui Chen, Lihao Liu, Kai Chen, Zijia Lin, Jungong Han, and Guiguang Ding. Yolov10: Real-time end-to-end object detection, 2024.

[48] Steven D Warren, Martin Alt, Keith D Olson, Severin DH Irl, Manuel J Steinbauer, and Anke Jentsch. The relationship between the spectral diversity of satellite imagery, habitat heterogeneity, and plant species richness. *Ecological Informatics*, 24:160–168, 2014.

[49] Qingfu Xiao and E Gregory McPherson. Tree health mapping with multispectral remote sensing data at uc davis, california. *Urban Ecosystems*, 8:349–361, 2005.

[50] Bogdan Zagajewski, Marcin Kluczek, Edwin Raczko, Ajda Njegovec, Anca Dabija, and Marlena Kycko. Comparison of random forest, support vector machines, and neural networks for post-disaster forest species mapping of the krkonoe/karkonosze transboundary biosphere reserve. *Remote. Sens.*, 13:2581, 2021.

[51] Haotian Zhao, Justin Morgenroth, Grant Pearse, and Jan Schindler. A systematic review of individual tree crown detection and delineation with convolutional neural networks (cnn). *Current Forestry Reports*, 9(3):149–170, 2023.

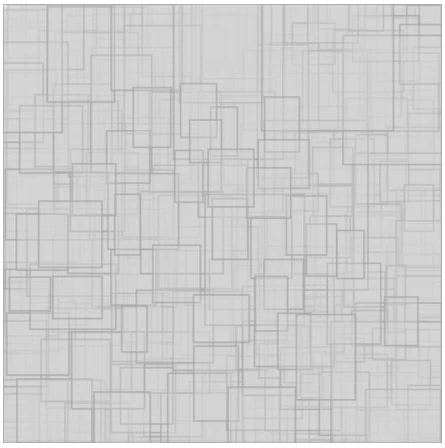
[52] Juepeng Zheng, Weijia Li, Maocai Xia, Runmin Dong, Haohuan Fu, and Shuai Yuan. Large-scale oil palm tree detection from high-resolution remote sensing images using faster-rcnn. In *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, pages 1422–1425, 2019.

Appendix A

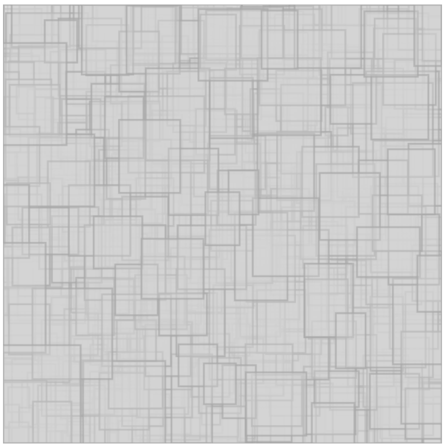
First appendix

A.1 Dataset Visualizations

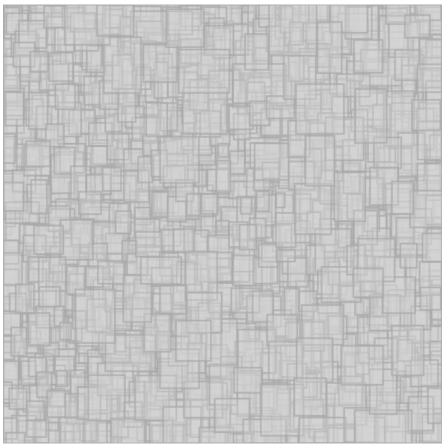
A.2 YOLOv10x - Extended results visualization



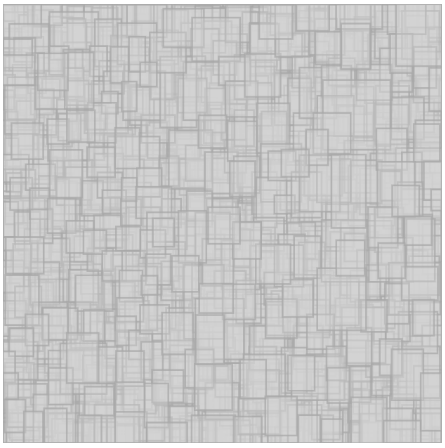
(a) Beech annotations



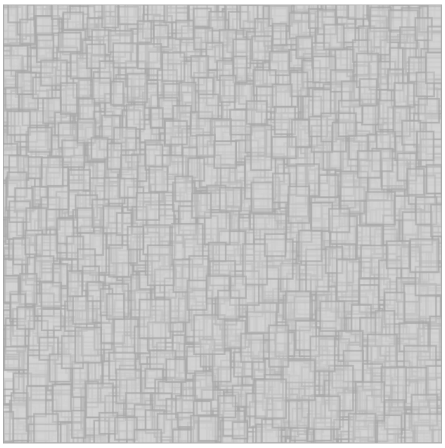
(b) Birch annotations



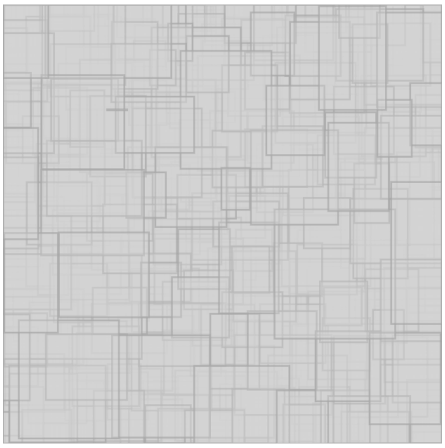
(c) Corsican pine annotations



(d) Douglas Fir annotations



(e) Norway Spruce annotations



(f) Oak annotations

Figure A.1: Bounding boxes and the locations in the image for different species.

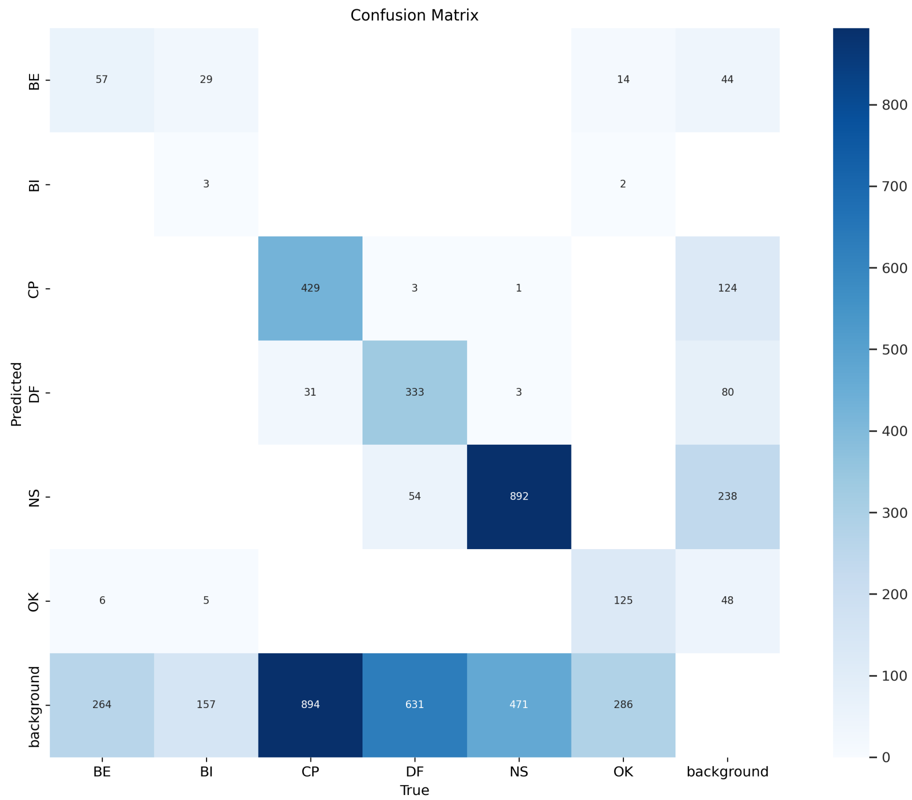


Figure A.2: YOLOv10x_Confusion matrix.

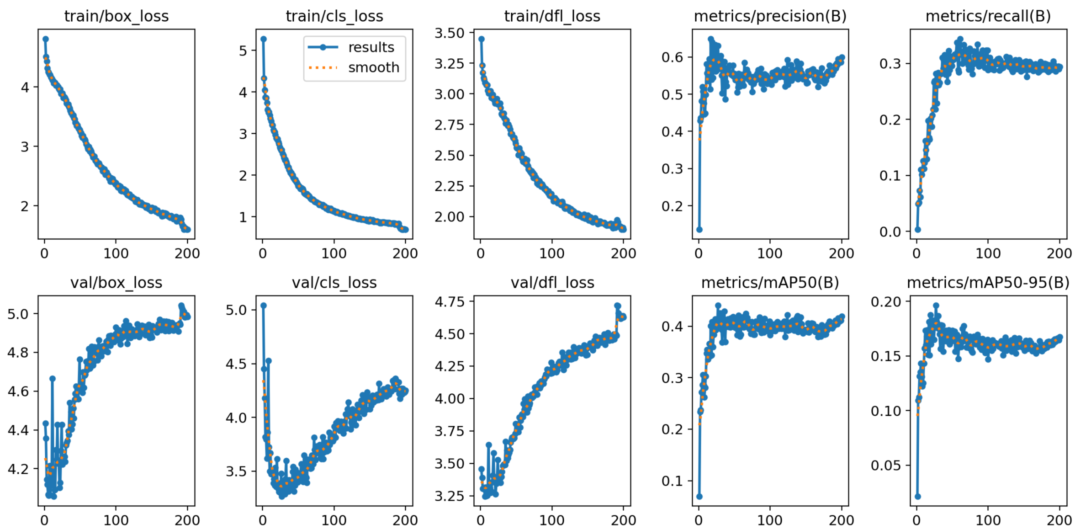


Figure A.3: YOLOv10x Loss Curves