

MapMyForest

Submitted in partial fulfillment of the requirements of the degree
**BACHELOR OF ENGINEERING IN COMPUTER
ENGINEERING**

By

Chinmay Desai Reg / D12C / 16

Atharva Deore Reg / D12C / 15

Gautam Rai Reg / D12C / 53

Shaanveer Singh Reg / D12C / 61

Name of the Mentor

Prof. Dr. Mrs. Gresha Bhatia



Vivekanand Education Society's Institute of Technology,

An Autonomous Institute affiliated to University of Mumbai

HAMC, Collector's Colony, Chembur,

Mumbai-400074

University of Mumbai (AY 2024-25)

CERTIFICATE

This is to certify that the Mini Project entitled "**MapMyForest**" is a bonafide work of
Chinmay Desai (16), Atharva Deore (15), Gautam Rai (53), Shaanveer Singh (61)
submitted to the University of Mumbai in partial fulfillment of the requirement for the
award of the degree of "**Bachelor of Engineering**" in "**Computer Engineering**".



10/10/21

(Prof. Dr. Mrs. Gresha Bhatia)

Mentor



(Prof. Nupur Giri)

Head of Department



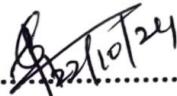
(Prof. Dr. J. M. Nair)

Principal
PRINCIPAL,
VIVEKANAND EDUCATION SOCIETY'S
INSTITUTE OF TECHNOLOGY
HASHU ADVANI MEMORIAL COMPLEX,
COLLECTOR'S COLONY, CHEMBUR,
MUMBAI-400 074, INDIA.

Mini Project Approval

This Mini Project entitled "MayMyForest" by **Chinmay Desai (16), Atharva Deore (15), Gautam Rai (53), Shaanveer Singh (61)** is approved for the degree of **Bachelor of Engineering in Computer Engineering.**

Examiners

1.....
(Internal Examiner Name & Sign)

2.....
(External Examiner name & Sign)

Date: 22 / 10 / 24

Place: Mumbai

Contents

Abstract	ii
Acknowledgments	iii
List of Abbreviations	iv
List of Figures	v
List of Tables	v
List of Symbols	vi
1 Introduction	1
1.1 Introduction	
1.2 Motivation	
1.3 Problem Statement & Objectives	
1.4 Organization of the Report	
2 Literature Survey	3
2.1 Survey of Existing System	
2.2 Limitation Existing system or Research gap	
2.3 Mini Project Contribution	
3 Proposed System	9
3.1 Introduction	
3.2 Architectural Framework	
3.3 Algorithm and Process Design	
3.4 Methodology	
3.5 Hardware & Software Specifications	
3.6 Experiment and Results for Validation and Verification	
3.7 Result Analysis and Discussion	
3.8 Conclusion and Future Work.	
References	28
4 Annexure	
4.1 Published Paper /Camera Ready Paper/ Business pitch/proof of concept (if any)	

Abstract

The **MapMyForest** project endeavors to create an innovative automated solution for tree enumeration and classification within forested areas, leveraging cutting-edge image processing techniques applied to satellite and aerial imagery. Traditional methods of tree monitoring, such as manual surveys and ground-based assessments, are often labor-intensive, time-consuming, and susceptible to inaccuracies, leading to challenges in data reliability and environmental decision-making. In response to these limitations, our project employs the YOLO (You Only Look Once) model, a state-of-the-art object detection algorithm, alongside Convolutional Neural Networks (CNN) for effective species classification.

By automating the processes of tree detection and categorization, the **MapMyForest** system significantly enhances the accuracy and efficiency of tree enumeration, thereby facilitating more reliable Environmental Impact Assessments (EIA) and ongoing forest monitoring. The software is designed to be user-friendly, providing stakeholders—including environmental agencies, non-governmental organizations (NGOs), and researchers—with real-time insights into forest ecosystems and their health. Additionally, the solution aims to lower monitoring costs and streamline the overall process of environmental conservation efforts. Ultimately, **MapMyForest** represents a vital step towards responsible land development and sustainable forestry practices, aligning with broader ecological goals and contributing to the preservation of biodiversity.

Acknowledgments

We are grateful to the individuals and organizations whose support made the completion of the MapMyForest project possible.

We extend our thanks to Dr. Gresha Bhatia for her guidance and assistance throughout the project. Her knowledge and direction were key in helping us address the challenges of tree enumeration and classification using image processing techniques.

We would also like to thank the Department of Computer Engineering at VESIT for their continuous support, as well as the college itself for providing the necessary resources and environment for our research.

Last but not least, we extend our collective appreciation to all team members who collaborated diligently on this project, bringing their unique expertise to the table. This research would not have been possible without the collective effort, dedication, and cooperation of the entire team. We sincerely thank every one of these contributors for their invaluable assistance in making this research project a reality.

LIST OF ABBREVIATIONS

1. AI: Artificial Intelligence
2. ANOVA: Analysis of Variance
3. API: Application Programming Interface
4. CNN: Convolutional Neural Network
5. DBN: Deep Belief Network
6. DL: Deep Learning
7. DNN: Deep Neural Network
8. EIA: Environmental Impact Assessment
9. EO: Earth Observation
10. GAN: Generative Adversarial Network
11. GIS: Geographic Information System
12. GPU: Graphics Processing Unit
13. GPS: Global Positioning System
14. Hyper-parameter: A parameter whose value is set before the learning process begins
15. IoT: Internet of Things
16. LSTM: Long Short-Term Memory
17. ML: Machine Learning
18. NLP: Natural Language Processing
19. NN: Neural Network
20. NDVI: Normalized Difference Vegetation Index
21. PCA: Principal Component Analysis
22. PNAS: Proceedings of the National Academy of Sciences
23. RNN: Recurrent Neural Network
24. SVM: Support Vector Machine
25. UAV: Unmanned Aerial Vehicle
26. YOLO: You Only Look Once
27. LLM: Large Language Model

LIST OF FIGURES

Figure Number	Figure Name	Page Number
3.2.1	Architectural Diagram	10
3.3.1	Modular Process Diagram	13
3.4.1	Yolo Diagram	17
3.6.1	Home Page	18
3.6.2	Authentication Page	21
3.6.3	Project Management Page	22
3.6.4	Image addition Page Layout	22
3.6.5	Enumerated Tree Page	23
3.6.6	Project Details and Analysis Layout	24

LIST OF TABLES

Table Number	Table	Page Number
2.1	Literature Review	4

LIST OF SYMBOLS

Symbol	Description
IOU	Intersection over Union
TP	True Positives
FP	False Positives
TN	True Negatives
FN	False Negatives
Precision	Proportion of correctly predicted objects out of all predicted objects
Recall	Proportion of correctly identified objects out of all ground truth objects
F1	F1 Score - Harmonic mean of Precision and Recall
mAP	Mean Average Precision across different IoU thresholds
AP	Average Precision for each class
k	Number of top classes considered in accuracy
p(r)	Precision as a function of recall
IoU \geq 0.5	Threshold for true positive detection

Introduction

1.1 Introduction

Tree enumeration in India poses a significant challenge due to the vast areas of land that need to be surveyed. Relying on manual labor for this task often leads to inaccurate data collection and analysis. Additionally, it becomes difficult to assess the types of trees being cut down for regional development, the environmental impact, and whether the species are endangered.

Our project, “MapMyForest”, offers a solution by automating the tree enumeration and classification process in forested areas through image analysis of aerial images using machine learning models. This project aims to revolutionize the Environmental Impact Assessment (EIA) process in forest conservation and development projects by providing accurate, real-time monitoring and detailed reporting.

The final product will cater to environmental agencies, NGOs, researchers, and other stakeholders involved in forestry and environmental sustainability.

1.2 Motivation

India’s goal of achieving net-zero carbon emissions by 2070 cannot be realized without the conservation of forests and a commitment to sustainable development. Forests play a crucial role in this process, but due to the large population, there is a growing need for space to provide basic facilities to people. This rapid development often leads to deforestation. Currently, tree enumeration is a complex, manual, and error-prone process, with no simpler or efficient methods available.

To address this challenge, we are developing “MapMyForest”, a machine learning-based web solution that automates tree enumeration and classification using image processing techniques. This solution will provide detailed analysis and help streamline the process of monitoring deforestation, aiding in sustainable development efforts.

1.3 Problem Statement & Objectives

Problem Statement:

Tree enumeration is a critical process conducted before developing land, especially in forested or environmentally sensitive areas. It involves assessing the potential environmental impact of deforestation or land development. The current methods, which rely heavily on manual labor, are extremely time-consuming, prone to errors, and require significant human effort in terms of time, physical labor, and resource management.

Even when accurate tree enumeration data is collected, the subsequent analysis is often inadequate. Key insights, such as the potential environmental impact, the disruption to the local ecosystem, and the effects on the biodiversity and life cycle in the region, are not

easily obtainable. Furthermore, developers need to comply with various environmental rules and regulations, which vary from one region to another. Gathering this information is a complex and time-intensive task, leading to delays in decision-making and project execution.

Objectives:

- **Automating Tree Enumeration:** Developing a machine learning-based solution using image analytics to automate tree enumeration in forest areas, enhancing accuracy and efficiency compared to traditional manual methods.
- **Environmental Impact Assessments (EIA):** Provide real-time EIA reports that offer insights into forest land changes, leveraging automated tree counting and species classification to improve environmental decision-making.
- **Deforestation Solutions:** Developing a solution that processes user-submitted images to assess the impact of deforestation and provides recommendations for mitigation. The system will also suggest or enforce legal regulations based on deforestation issues specific to the region, according to the laws and policies in place in that country.
- **Regulatory Compliance:** Integrate local environmental laws and regulations into the platform to ensure developers are aware of the legal requirements and obligations before proceeding with land development.

1.4 Organization of the Report

The report is organized as follows: The Literature Survey section reviews the research conducted in similar fields, focusing on their methodologies and limitations. Next, we analyze the existing systems currently used to address the problem, highlighting their limitations and identifying the issues our project aims to solve. The Proposed System provides details about the conceptual design, architectural framework, and methodology we will implement to address the problem. Ultimately results presents the outcomes of the experiments which including model accuracy and analysis. Finally, the Conclusion and Future Work discusses potential improvements and future directions to enhance the project's scope and effectiveness.

2. Literature Survey

A literature survey is a crucial part of academic research that provides a comprehensive overview of existing knowledge on a specific topic. It involves analyzing and synthesizing information from various credible sources, such as scholarly articles, books, and conference papers, to understand the current state of research in the field. By reviewing previous work, it identifies key theories, methodologies, and findings, while also highlighting gaps or inconsistencies that may require further exploration. This process helps researchers contextualize their study within the broader research landscape, demonstrating how their work contributes to or challenges existing knowledge. Additionally, it ensures that new research builds on prior efforts rather than duplicating them, saving time and resources.

1. Satellite Image Analytics for Tree Enumeration for Diversion of Forest Land (2024)

Abstract:

This paper presents the development of a software solution for automating tree enumeration using satellite imagery and advanced image analytics. Traditional methods for tree enumeration in land diversion assessments are labor-intensive and error-prone. By integrating Geographic Information Systems (GIS) with Convolutional Neural Networks (CNNs), the system detects and classifies trees based on features such as texture, color, and shape. PostgreSQL with PostGIS ensures robust spatial data management for tracking deforestation and conducting environmental impact assessments (EIA). This automation enhances the speed, accuracy, and cost-effectiveness of forest monitoring, making it a valuable tool for environmental agencies and developers. The project contributes to sustainable forestry management, providing real-time insights for better decision-making.

Inference:

This paper demonstrates how integrating GIS with machine learning models can significantly improve the accuracy and efficiency of tree enumeration in forest land diversion assessments, supporting sustainable environmental practices.

2. Intelligent Forest Assessment: Advanced Tree Detection and Enumeration with AI (2024)

Abstract:

This paper discusses the development of an advanced system that automates tree detection and enumeration using the YOLOv8 model. The system aims to enhance environmental monitoring by providing a scalable solution for tree detection across various environmental conditions. Using Roboflow annotations, the system optimizes dataset structuring and training to achieve high accuracy. It categorizes trees based on height, making it useful for detailed forest monitoring. The system's accuracy exceeds 90%, and it automates the traditional manual process of tree enumeration, improving forest management and environmental decision-making.

Inference:

The paper highlights the potential of machine learning models like YOLOv8 to revolutionize environmental monitoring by automating tree detection and improving the efficiency of forest conservation efforts.

3. Deep Learning Enables Image-Based Tree Counting, Crown Segmentation, and Height Prediction at National Scale (2023)

Abstract:

This paper introduces a deep learning framework for tree counting, crown segmentation, and height prediction using aerial imagery. Applied at a national scale in Denmark, the model identified large trees and showed that 30% of tree cover exists outside forests. Despite minor challenges in detecting smaller trees, the framework offers a flexible solution for tree resource management across different landscapes. The system creates digitalized tree databases that contribute to forest management, carbon stock assessments, and climate change mitigation.

Inference:

The integration of deep learning in tree resource management provides comprehensive data at a national level, which is essential for forest management, environmental policy planning, and sustainability efforts.

4. Data Augmentation and Few-Shot Change Detection in Forest Remote Sensing (2023)

Abstract:

This paper presents an innovative framework for detecting forest changes using data augmentation and few-shot learning approaches in remote sensing. The model incorporates convolutional neural networks (CNN) and deep convolutional generative adversarial networks (DCGAN) to generate synthetic forest data, improving model training and balancing data samples. The framework enhances change detection performance, achieving a 91% F1 score, offering a promising solution for automating forest monitoring.

Inference:

By combining data augmentation with few-shot learning, this paper provides an effective method for improving forest change detection with limited labeled data, expanding the possibilities for remote sensing applications in environmental monitoring.

5. Real-Time Algorithm for Tree Detection, Recognition, and Counting in a Video Sequence (2020)

Abstract:

This paper proposes a real-time system for tree detection, recognition, and counting in UAV-based video sequences. The system uses Kalman filtering and the Hungarian algorithm to address challenges like motion blur and occlusions, achieving detection accuracy of 92.9% and counting accuracy of 91.2%. The algorithm enhances vegetation segmentation, enabling detailed mapping of tree attributes such as location and health. It has applications in large-scale agricultural and environmental monitoring, contributing to precision agriculture and sustainable land management.

Inference:

This system provides an efficient real-time solution for automated tree counting and environmental monitoring using UAV imagery, demonstrating significant potential for precision agriculture and large-scale forestry applications.

6. Computer Vision System for Automatic Counting of Planting Microsites Using UAV Imagery (2019)

Abstract:

The paper presents an automated system using UAV imagery and computer vision techniques to detect and count planting microsites. By combining Local Binary Patterns (LBP) and Convolutional Neural Networks (CNN), the method enhances efficiency and accuracy in forestry operations, even under challenging environmental conditions.

Inference:

The study demonstrates a robust system for automatic microsite detection in forestry operations, achieving improved accuracy over manual counting methods and offering a faster, more adaptable solution for forestry management.

7. Detection and Enumeration of Trees using Cartosat2 High-Resolution Satellite Imagery (2018)

Abstract:

This paper presents a method for automated tree detection and counting using Cartosat2 satellite imagery. It employs Contrast Limited Adaptive Histogram Equalization (CLAHE) for image enhancement, Canny edge detection for segmentation, and contour detection for feature extraction, automating the tree counting process for forestry and urban planning applications.

Inference:

The method offers high accuracy in tree detection with an average deviation of 1.948% from manual counts. Its effective use of CLAHE and contour-based segmentation is validated on Cartosat2 imagery, proving valuable for forestry and urban applications.

8. Fast Visual Object Counting via Example-Based Density Estimation (2016)

Abstract:

The paper proposes a fast density estimation-based visual object counting (DE-VOC) method that improves efficiency without sacrificing effectiveness. Using example-based density estimation and locally linear embedding, the method estimates object counts from images, maintaining comparable accuracy to mainstream approaches while significantly reducing testing time.

Inference:

This method provides a fast and effective solution for visual object counting, achieving high accuracy with lower computational costs, though it requires careful cluster number selection for optimal performance.

9. UAS-based Automatic Bird Count of a Common Gull Colony (2013)

Abstract:

This study presents a method for automatically counting birds in a common gull colony using high-resolution imagery from unmanned aircraft systems (UAS). The approach uses image classification, GIS post-processing, and 3D point cloud analysis to detect and count birds while analyzing their habitat preferences.

Inference:

The automated bird counting system achieved $\pm 3\%$ accuracy with no significant disturbance to the colony. It provided valuable insights into bird nesting preferences by analyzing vegetation height and terrain features via 3D point clouds.

2.1 Survey of Existing Systems

The current system for tree enumeration predominantly relies on manual processes. Typically, a team of individuals is assigned a specific area where they manually mark each tree with a number. Afterward, the trees are often cut down, and a basic count is performed to understand the total number of trees. However, this system lacks any meaningful analysis of the environmental impact caused by tree removal. Most of the time, no effort is made to assess the broader ecological consequences, such as the effects on biodiversity, soil erosion, or carbon sequestration. Furthermore, people involved in these processes are often unaware of the legal frameworks and environmental regulations that should be followed during such operations.

Manual methods also pose several significant challenges. Managing labor can be cumbersome, and costs tend to rise, especially when dealing with large or difficult terrains, such as mangrove forests or mountainous regions. These areas present additional risks to the safety of workers, potentially putting their lives in danger due to difficult accessibility or environmental hazards.

In terms of software solutions, the available systems do not provide comprehensive end-to-end solutions tailored to India's specific needs. Most software is designed for use in other countries, and as a result, they incorporate environmental laws, cost structures, and analysis models that are relevant to their specific regions. This makes them less effective or impractical for deployment in India, where different regulations and ecological concerns must be taken into account.

2.2 Limitation of Existing Systems

While the existing systems for tree enumeration and environmental analysis offer some level of automation, they present several critical limitations, especially when applied in the context of India's unique environmental and legal landscape:

1. Accuracy and Precision:

Many of the existing systems fail to provide reliable accuracy metrics for tree enumeration. This leads to inaccuracies in data collection and makes it difficult to conduct reliable environmental assessments.

2. Lack of Environmental Impact Analysis:

Current systems tend to focus solely on counting and basic classification without integrating tools for assessing the environmental impact of tree removal. Key factors such as biodiversity loss, carbon emissions, and changes to local ecosystems are not considered, limiting the usefulness of the data for conservation efforts or sustainable forestry practices.

3. Scalability Issues:

Many existing solutions are computationally expensive and not designed to handle large-scale operations. This poses a challenge when monitoring vast forest regions, such as national parks, biodiversity hotspots, or areas with rapidly changing environments. As a result, the cost and time required to scale these systems make them impractical for large-scale deployment.

4. Regional and Legal Incompatibility:

Most available software solutions are designed for regions outside of India, incorporating country-specific regulations, environmental laws, and operational methodologies that do not align with India's legal frameworks. This makes it difficult to ensure compliance with local environmental laws and undermines the system's relevance in the Indian context. Moreover, these tools often fail to account for the unique characteristics of Indian forests, such as the presence of mangroves, which require specialized handling and consideration.

5. Safety and Accessibility:

Manual methods, especially in hard-to-reach areas like mangroves or rugged terrain, can expose workers to dangerous situations. The absence of automated systems that can handle these challenging environments without risking human lives remains a significant drawback of existing systems.

6. Limited Customization:

Existing solutions lack flexibility and customization options tailored to the diverse forest types in India. The inability to adjust models or workflows for different types of ecosystems or varying environmental conditions reduces the adaptability of these tools for widespread use in the country.

2.3 Mini Project Contribution

The "MapMyForest" project addresses these gaps by providing features like:

- Automated Tree Enumeration and Classification: The core contribution of this project lies in the development of an advanced machine-learning model that automates the process of tree enumeration and classification. By leveraging image analytics from aerial imagery, the system will accurately count trees, classify species, and analyze forest density. This automated approach aims to significantly reduce the reliance on manual labor, ensuring that forest data is gathered efficiently and accurately.
- Real-Time Environmental Impact Assessment (EIA): Another key aspect of the project is the creation of a real-time EIA module, designed to provide insightful reports on the environmental impact of deforestation and land use changes. This module will process user-submitted images to evaluate the extent of forest loss, tree species affected, and the broader ecological implications. The EIA module serves to enhance environmental monitoring efforts by offering stakeholders detailed, data-driven insights into the condition of forested areas.
- Seamless Video Classification System: In addition to image analysis, the project incorporates a video classification system. Users can submit videos of forested areas, and the system will analyze them using machine learning algorithms to detect and classify trees, identify patterns of deforestation, and generate relevant data reports. This feature broadens the scope of data collection, allowing for the inclusion of dynamic, real-time footage in environmental assessments.
- User Interface for Efficient Data Submission: The platform will feature a user-friendly interface that facilitates the submission of aerial images and videos for analysis. Recognizing the complexity of forest data collection, the interface is designed to be intuitive and accessible, allowing users to upload media files and access the analysis results with ease. This interface prioritizes user convenience while maintaining the technical sophistication necessary for accurate data processing.
- Integration of Regulatory Compliance: The project will also incorporate a regulatory compliance framework, ensuring that users are informed about the environmental laws and regulations related to tree cutting and land development. This feature will provide guidance on legal procedures, helping users align their activities with local, regional, and national environmental policies.

3. Proposed System

3.1 Introduction

The proposed system aims to automate the process of tree enumeration and classification using advanced machine-learning techniques on satellite and aerial imagery. The system leverages the YOLO (You Only Look Once) object detection algorithm to detect trees in large forest areas, while a Convolutional Neural Network (CNN) model is employed for the classification of tree species based on their visual characteristics. This combination ensures both accurate detection and reliable species categorization.

The platform provides a streamlined and user-friendly interface that enables environmental agencies, researchers, and other stakeholders to upload image data, access analysis results, and generate reports. By automating tree counting and classification, the system addresses the inefficiencies of traditional manual methods, offering a scalable, efficient, and accurate solution for forest management and conservation efforts.

In addition to tree detection and classification, the system integrates real-time environmental impact assessment (EIA) capabilities, providing stakeholders with data-driven insights for decision-making related to land development and forest conservation. The platform will also ensure regulatory compliance by integrating local environmental laws, helping users to adhere to legal requirements while carrying out forest-related activities.

3.2 Architectural Framework and Modular Diagram

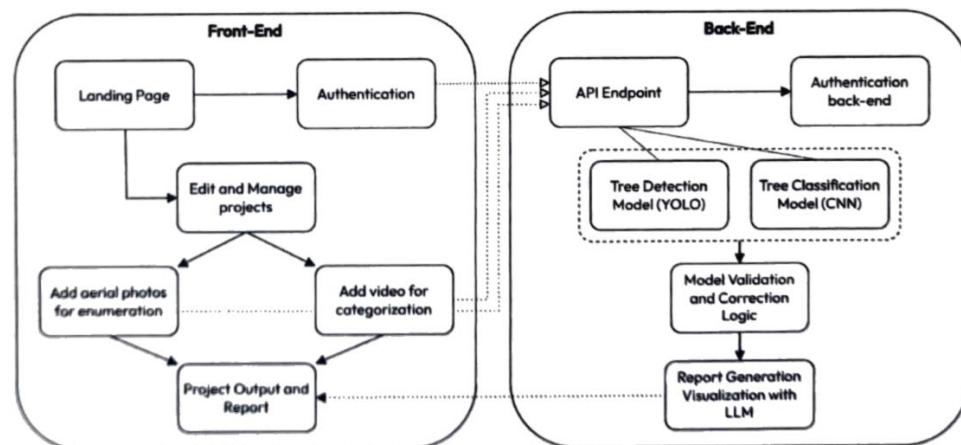


Fig 3.2.1 : Architectural Diagram

This architectural diagram outlines a system divided into two primary components: the front end and the back end. Each component interacts with various elements to fulfill the goals of the system, which seem to center around managing and analyzing aerial photos and videos, specifically for tree detection and classification.

Front-End Breakdown:

1. Landing Page:
 - This is the entry point of the application. Users will start here before proceeding to other parts of the system. It likely serves as a navigational hub or provides an overview of the project.
2. Authentication:
 - From the landing page, users proceed to authentication. This is necessary to improve data privacy and ownership to the system's functionalities. It could involve user login, registration, or even session handling.
 - This step is critical for security, ensuring only authorized users can access sensitive features like managing projects or uploading data.
3. Edit and Manage Projects:
 - Once authenticated, users can access this module. Here, users can create, edit, or manage various projects, such as uploading or viewing aerial photos or videos related to the enumeration (counting) and categorization of trees.
4. Add Aerial Photos for Enumeration:
 - Users can add aerial photos here, which are essential for the enumeration process (tree counting). These photos will be processed on the back end for tree detection using a specialized model (more details below).
5. Add Video for Categorization:
 - This module allows users to upload videos, which will be used for the categorization process determining tree species or health. Like aerial photos, these videos are sent to the back-end for processing.
6. Project Output and Report:
 - After processing aerial photos and videos, the system generates outputs and reports. The output could be in the form of visualizations or detailed data about tree enumeration and categorization. This module aggregates the user's results.

Back-End Breakdown:

1. API Endpoint:
 - This is the connection between the front-end and back-end. All user data, like uploaded photos and videos, passes through this API for further processing.

It also facilitates communication between the user interface and the machine learning models.

2. Authentication Back-End:

- This handles the actual logic of user authentication (login/logout, password validation, token generation, etc.). It ensures that only verified users can interact with the complete functionalities of the website through the API and access back-end services.

3. Tree Detection Model (YOLOv11):

- YOLO (You Only Look Once) is a real-time object detection model. In this context, it will be used for detecting trees in aerial photos. YOLO's strength lies in its ability to perform fast and accurate detections, making it well-suited for large-scale image analysis like tree detection from aerial shots.

4. Tree Classification Model (CNN):

- This component involves a Convolutional Neural Network (CNN), a type of deep learning model widely used for image classification tasks. It is likely used to categorize the detected trees, such as identifying tree species, health status, or other characteristics based on video data uploaded in the front end.

5. Model Validation and Correction Logic:

- After the tree enumeration and tree classification models process the photos and videos, this module ensures the results are accurate. This step involves verifying the predictions from the models (detection and classification) against ground truth data or user input, allowing for corrections before report generation.

6. Report Generation and Visualization with LLM:

- This module uses an LLM (Large Language Model) to generate a comprehensive report based on the analysis results. The LLM summarizes findings, provides insights, or visualizes the data, integrating natural language descriptions to make the reports user-friendly.

Connections Between Components:

- **Dotted Lines:**

- These lines represent communication paths or data flows between modules. For instance, the connection between the authentication system on the front-end and back-end ensures that requests from the user are properly authenticated before proceeding with project management or data uploads.

- The connection between "Add aerial photos for enumeration" and the back-end shows that these photos are processed using the YOLO model, while videos uploaded for categorization are processed using the CNN model.
- The final dotted lines from the "Edit and Manage projects" and the "Project Output and Report" show the flow of results, as the processed data (trees detected and categorized) is returned to the front-end for reporting.

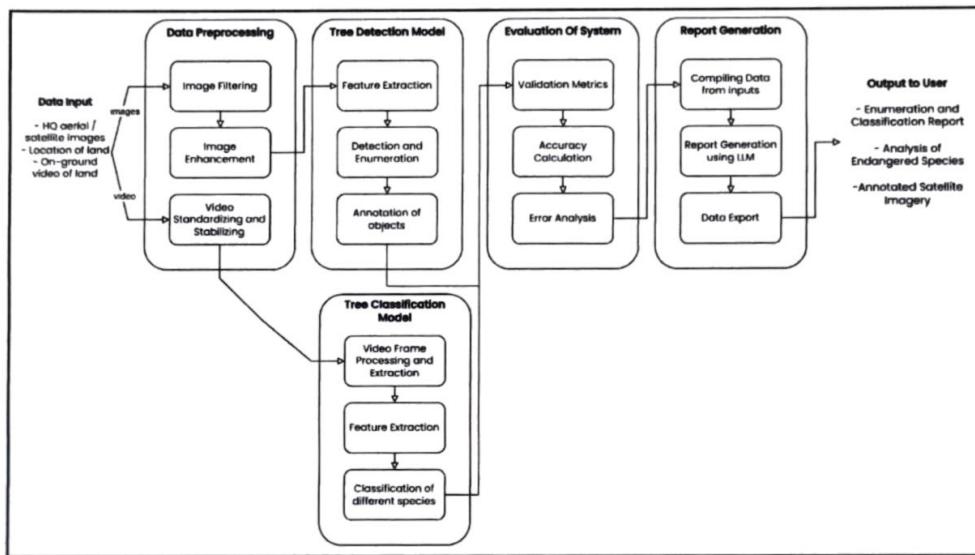


Fig 3.3.1: Modular Process Diagram

The diagram provides a detailed flow for the algorithmic and process design of a system that processes aerial/satellite images and video footage to detect and classify trees. The system aims to generate reports on tree enumeration, classification, and possibly endangered species analysis, with annotated satellite imagery for users. The process is divided into distinct phases, each serving a specific role within the system.

1. Data Input:

- The system takes in different types of data as input:
 - HQ aerial/satellite images: High-quality imagery from satellites or aerial drones, likely covering large areas of forest or land.
 - Location of land: The geospatial data that tags the location of the land under analysis.
 - On-ground video of land: Video footage captured on the ground, which is likely used for fine-grained tree classification, such as identifying species or health conditions.

- The incoming data is then separated into two streams: images (aerial/satellite images) and videos (on-ground footage).

2. Data Preprocessing:

- This stage is crucial for preparing the raw input data for further analysis by the machine learning models.
- Image Filtering:
 - Raw images may contain noise, distortions, or irrelevant information. Filtering helps clean up the data by removing noise, sharpening features, or adjusting contrast.
- Image Enhancement:
 - Further improves image quality by adjusting color, brightness, and resolution. This ensures that the tree detection model has clear, high-quality images to work with, leading to more accurate detection.
- Video Standardizing and Stabilizing:
 - Videos can be shaky or captured in inconsistent conditions (like different lighting or angles). Standardizing video ensures a uniform frame rate and resolution, while stabilization removes shaking or jitters. This step is critical for consistent tree classification.

3. Tree Detection Model:

- Once the data is preprocessed, it is sent to the detection model, which performs several tasks:
- Feature Extraction:
 - The system identifies key features in the aerial/satellite images that are essential for detecting trees. Feature extraction simplifies the image by focusing on relevant details like edges, textures, and shapes, making it easier to identify trees.
- Detection and Enumeration:
 - The system detects and counts individual trees within the image using a machine-learning model such as YOLO. This step involves identifying distinct tree objects in the image and recording their count.
- Annotation of Objects:
 - Detected trees are then annotated, meaning each tree is marked with metadata such as its position or any other relevant features. This allows further steps, like classification, to know exactly where each tree is located in the image.

4. Tree Classification Model:

- Parallel to tree detection, the on-ground videos are processed through the classification model, which focuses on more detailed tree identification:
- Video Frame Processing and Extraction:
 - The input video is split into individual frames, and relevant data is extracted from each frame to capture important visual information. Video data is processed differently from static images, often requiring more computational power.
- Feature Extraction:
 - Similar to the feature extraction in the detection model, the video frames are analyzed to identify features that can help in classification, such as leaf patterns, tree bark textures, or growth structures.
- Classification of Different Species:
 - Using the extracted features, the model classifies each tree into species or other categories, such as health status or growth stage. The classification model likely uses a deep learning architecture like a CNN (Convolutional Neural Network) to perform this task.

5. Evaluation of System:

- After the tree detection and classification models complete their tasks, the results are passed to an evaluation module to ensure accuracy and reliability.
- Validation Metrics:
 - The results from the detection and classification models are validated against known metrics. This could include comparing the model's output with ground-truth data to measure precision, recall, and other performance metrics.
- Accuracy Calculation:
 - A critical step in assessing how well the models performed in terms of detecting and classifying trees. If the models have a high error rate, they may need further training or adjustment to improve their accuracy.
- Error Analysis:
 - This step identifies any issues or inaccuracies in the system, such as false detections, missed trees, or incorrect classifications. It provides insight into what went wrong and how to improve the models.

6. Report Generation:

- After validation and error analysis, the system moves to generate a report that compiles all the information collected from the data inputs.
- Compiling Data from Inputs:
 - Data from the detection and classification models is compiled into a structured format. This could include the number of trees detected, their species classification, and their location on the map.

- Report Generation using LLM:
 - A large language model (LLM) is used to generate a human-readable report based on the compiled data. The LLM might summarize findings, highlight key insights, or even suggest further actions based on the results of the analysis.
- Data Export:
 - The final report is exported, possibly in formats like PDF or CSV, to be shared with the user or integrated into other systems.

7. Output to User:

- The final outputs are delivered to the user in various forms:
 - Enumeration and Classification Report: A detailed report summarizing the number of trees detected and their classification.
 - Analysis of Endangered Species: If any endangered tree species are detected, the system highlights them for conservation or legal purposes.
 - Annotated Satellite Imagery: The processed and annotated satellite imagery, showing detected trees and their corresponding classifications.

3.3 Algorithm and Process Design

Tree Enumeration Algorithm:

```

FUNCTION TreeEnumeration(imageryData):
  FOR each image IN imageryData:
    imageChunks ← Divide image into smaller chunks
    preprocessedImage ← Resize(imageChunks, 640x640)
    augmentedImage ← ApplyDataAugmentation(preprocessedImage)

    model ← LoadYOLOv11Model()
    results ← []

    FOR each chunk IN augmentedImage:
      results ← ParallelEnumerate(chunk)

    totalTrees ← COUNT(results)
    RETURN totalTrees, results.images
  
```

```

FUNCTION ParallelEnumerate(imageryData):
  batches ← DivideIntoBatches(imageryData)
  parallelResults ← []
  
```

```

  FOR each batch IN batches:
    result ← ProcessBatchConcurrently(batch)
  
```

```
parallelResults.Append(result)

combinedResults ← MergeResults(parallelResults)
RETURN combinedResults
```

1. TreeEnumeration Function

1. **Input:** Imagery data (a set of images) is provided by the user.
2. **Step 1:** Loop through each image in the imageryData:
 - o Each image is divided into smaller chunks for easier processing.
3. **Step 2:** Each chunk is resized to a consistent resolution of 640x640 pixels.
4. **Step 3:** Data augmentation techniques (e.g., rotation, flipping) are applied to enhance the variety of the training data. This step ensures better model generalization.
5. **Step 4:** Load the pre-trained YOLOv11 model for tree enumeration.
6. **Step 5:** Create an empty list called results to store the tree detection results.
7. **Step 6:** Loop through each preprocessed and augmented image chunk:
 - o Call the ParallelEnumerate function to process each chunk in parallel for faster inference.
8. **Step 7:** Once all chunks are processed, count the total number of trees from the detection results.
9. **Output:** Return the total tree count and the image results showing tree locations.

2. ParallelEnumerate Function

1. **Input:** The preprocessed imagery data (image chunks) from the TreeEnumeration function.
2. **Step 1:** The imagery data is divided into multiple smaller batches.
3. **Step 2:** Create an empty list called parallelResults to store the results from each batch.
4. **Step 3:** Loop through each batch:
 - o Each batch is processed concurrently using multiple GPUs or CPU cores. This speeds up the inference process.
5. **Step 4:** Append the results of each batch to parallelResults.
6. **Step 5:** Merge the results from all batches to form the combined results.
7. **Output:** Return the merged results from the parallel processing.

The TreeEnumeration function begins by processing the input imagery data, which consists of high-resolution satellite or drone images. To handle these large images efficiently, the function first divides each image into smaller, manageable chunks. These chunks are then resized to a standard resolution of 640x640 pixels to ensure consistency when passed into the YOLOv11 model. Data augmentation techniques are applied to these preprocessed chunks to increase the diversity of the training data and enhance the model's performance.

Once the images are preprocessed, the function loads the YOLOv11 model, which is responsible for detecting trees within the image chunks. However, instead of processing each image sequentially, the function leverages parallel processing for efficiency. It calls the ParallelEnumerate function, which divides the image chunks into batches and processes them concurrently. This allows multiple chunks to be processed simultaneously, speeding up the tree enumeration process. The results from all batches are combined and the total number of trees is counted. Finally, the function returns the total tree count along with the visual results of tree detections on the images.

The ParallelEnumerate function handles the parallel processing aspect. It splits the imagery data into batches and processes each batch concurrently across multiple processing units (CPUs or GPUs). This distributed processing approach significantly reduces the overall computation time, especially when dealing with large datasets. After processing each batch, the results are merged to produce a comprehensive output that is used in the final enumeration of trees.

3.4 Methodology

Overview

The MapMyForest project aims to provide an automated platform for landowners to assess the forest cover on their land. By uploading drone, aerial, or satellite imagery, users can enumerate trees on their land and identify the variety of tree species present. This system leverages deep learning models, including YOLOv11 for tree enumeration model and a Convolutional Neural Network (CNN) for the species identification model. Finally, the output of both models is converted to a comprehensive analysis using an LLM.

Data Collection

MapMyForest requires users to upload two types of data:

- **Imagery:** High-resolution images of forested land are captured using drones, aerial platforms, or satellites sourced by the user. These images are uploaded on MapMyForest and cover various regions of the user's land.
- **Video Footage:** Users can also upload video tours of the forest. The video provides dynamic views of the landscape, offering more detailed insights into the flora.

Tree Enumeration Using YOLOv11

Tree counting is accomplished using the YOLOv11 model, which is trained on annotated satellite and drone imagery datasets. The model detects individual trees within each image chunk and calculates the total number of trees in the uploaded imagery.

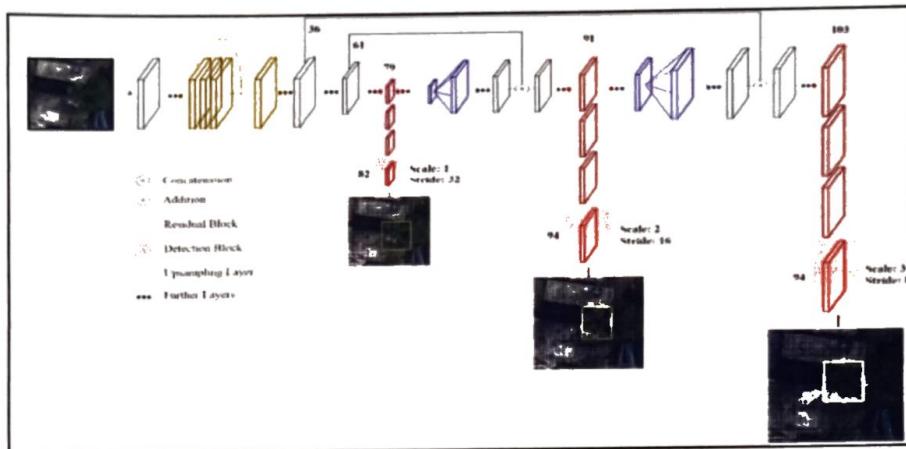


Fig 3.4.1 : Yolo Diagram

(source:- <https://www.researchgate.net/publication/354042454/figure/fig2/>
AS:1059088695820288@1629517830615/The-custom-YOLoV3-Architecture-for-coconut-tree-detection.jpg)

YOLOv11 is an evolution of the original YOLO (You Only Look Once) model, designed for real-time object detection. It uses a fully convolutional architecture, with improvements in bounding box regression, multi-scale feature extraction, and anchor-free object localization. The model is particularly suited for detecting objects in high-resolution aerial or satellite imagery due to its ability to process large images with high accuracy.

Training

The YOLOv11 model was pre-trained on large-scale object detection datasets such as COCO and then fine-tuned using custom-labeled aerial and satellite imagery of forests. The training data was annotated manually, where each tree in the images was labeled with a bounding box. All images were resized to a 640px x 640px resolution for consistency. Data augmentation techniques such as rotation, flipping, and random cropping were applied to increase the diversity of training data.

Inference and Post-Processing

Once the landowner uploads the imagery, the YOLOv11 model performs inference, detecting and counting individual trees. Post-processing filters, including non-max suppression, are applied to eliminate overlapping detections and enhance accuracy. The final count is displayed to the user, along with a visual representation of tree locations.

Tree Species Identification Using CNN

For species identification, the system relies on a CNN model to classify tree species from the uploaded video footage.

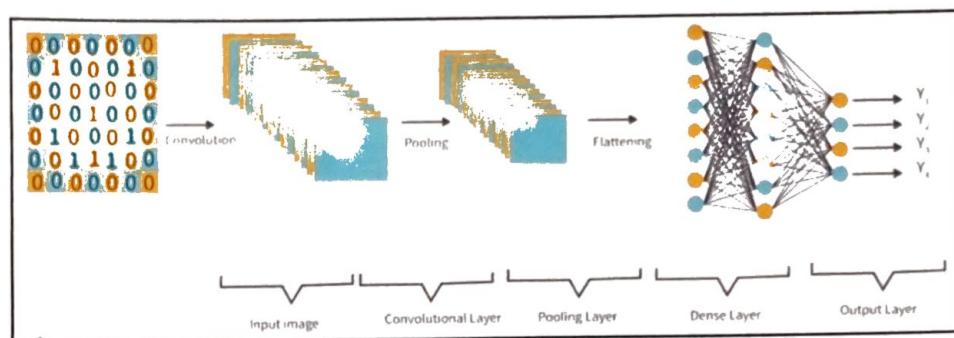


Fig 3.4.2: CNN Architectural Diagram
(source:- www.interviewbit.com/blog/wp-content/uploads/2022/06/Typical-CNN-Architecture-800x292.png)

Model Architecture

The CNN used for species identification consists of multiple convolutional layers followed by max-pooling layers, which progressively reduce the spatial dimensions while preserving key features. Fully connected layers at the end of the network output the probabilities of different tree species.

Training

The CNN was trained on a diverse dataset of tree species images, containing over 100 species of trees. Each frame of the video is extracted and passed through the CNN for a species classification.

Inference

During inference, the uploaded video is split into individual frames, and the CNN processes each frame to identify tree species. A majority voting mechanism is applied to aggregate predictions across multiple frames, resulting in a robust species classification output. The final result provides the user with a list of identified species along with their respective confidence scores.

Evaluation Metrics

Both models were evaluated using industry-standard metrics:

- **For YOLOv11 Tree Enumeration:**

1. **Intersection over Union (IoU)**

- o **IoU** is a measure of how much the predicted bounding box overlaps with the ground truth bounding box. It's calculated as:

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

- IoU values range from 0 to 1, where 1 means perfect overlap. A threshold (e.g., $\text{IoU} \geq 0.5$) is usually set to determine if a predicted box is considered a true positive.

2. Precision and Recall

- **Precision** measures the proportion of correctly identified objects out of all the objects predicted by the model:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

- **Recall** measures the proportion of correctly identified objects out of all ground truth objects:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

3. Mean Average Precision (mAP)

- **mAP** is the most commonly used metric for object detection models like YOLO. It calculates the average precision (AP) for each class over different IoU thresholds (usually 0.5 to 0.95) and takes the mean of all APs.

$$AP = \int_0^1 p(r) dr$$

- mAP is then the mean of APs for all object classes, representing the model's overall detection performance.

4. F1 Score

- The **F1 score** balances precision and recall to give a single metric:

$$F1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

• For CNN Species Identification:

- **Accuracy** is the most straightforward metric used for CNNs in classification tasks. It measures the proportion of correctly predicted classes out of the total predictions:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Predictions}}$$

- **Top-k Accuracy**

In cases where a CNN might output probabilities for multiple classes, top-k accuracy is used. It measures how often the true class is within the top k predicted classes (commonly k=5).

$$\text{Top 5 Accuracy} = \frac{\text{Number of Correct Top 5 Predictions}}{\text{Total Predictions}}$$

- **Confusion Matrix**

A **confusion matrix** is a table that breaks down the true positives, true negatives, false positives, and false negatives for each class. It provides insight into where the model is making errors:

- **True Positives (TP):** Correctly predicted class instances.
- **False Positives (FP):** Incorrectly predicted as the class.
- **True Negatives (TN):** Correctly predicted as not belonging to the class.
- **False Negatives (FN):** Incorrectly predicted as not belonging to the class when it should have been.

LLM-assisted Report Generation

MapMyForest integrates Large Language Model (LLM) technology to assist in generating detailed, user-friendly reports based on the outputs of both the YOLOv11 tree enumeration model and the CNN species identification model. Once the models complete their tasks, the LLM processes the raw data, including tree counts, detected species, and their respective confidence scores, to generate a comprehensive summary.

This summary is structured in a natural, easy-to-understand language, highlighting key findings such as the total number of trees, the diversity of species identified, and any notable patterns or insights. Additionally, the LLM customizes the report based on user preferences, adding visual aids like charts or maps where applicable, and providing actionable recommendations, such as forest management tips.

Batch Parallel Processing

To optimize the processing time for large datasets, MapMyForest employs batch parallel processing, which allows the system to divide uploaded imagery into smaller chunks that are processed concurrently across multiple GPUs or CPU cores. This parallel execution ensures that each batch of images is handled independently and simultaneously, significantly reducing the overall inference time for tree detection.

By leveraging distributed computing frameworks like Apache Spark or Dask, the system dynamically allocates resources based on demand, enabling horizontal scaling and optimal use of computational power. This approach not only accelerates the processing of large, high-resolution datasets but also enhances user experience by minimizing wait times for tree enumeration results.

3.5 Hardware & Software Specifications

Hardware:

- **Developer Specs:** Intel i5 (10th gen or newer) / AMD Ryzen 5, 16GB RAM, 512GB SSD or more.
- **User Specs:** Intel i3 (10th gen or newer), 8GB RAM, 264 GB SSD or more.

Software:

- **Computer Vision Libraries:** OpenCV (v4.8.0), TensorFlow (v2.14.0), PyTorch (v2.0.1).
- **Image Processing Tools:** QGIS (v3.30.3), ArcGIS (v3.1.0).
- **Frontend Development:** React (v18.2.0), Material UI (v5.14.8).
- **Backend Development:** Node.js (v20.5.1), Express (v4.18.2), Flask (v2.0.1), MongoDB (v8.0).

Tools:

- **Imagery Sources:** Google Earth Engine, Sentinel Hub.
- **Computer Vision Tool:** Rovren Flow.
- **Design Tools:** Figma, Canva.

3.6 Results



Fig 3.6.1 : Home Page

The MapMyForest landing page highlights the platform's focus on tracking and analyzing tree data using aerial imagery, with a clean layout and navigation bar for easy access to different sections.

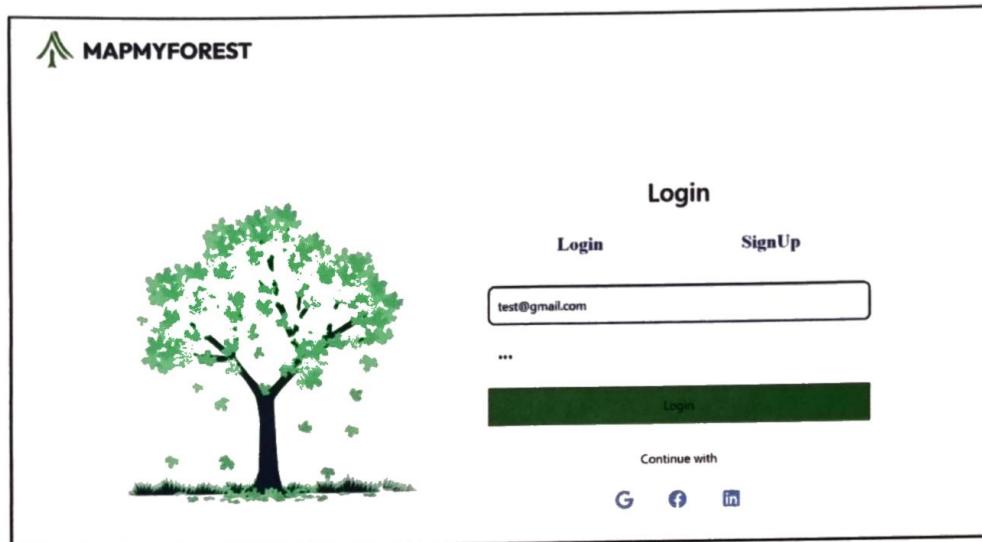


Fig 3.6.2 : Authentication Page

This is the authentication page for "MapMyForest," providing a secure, end-to-end authentication process. Users can log in using their email credentials or through third-party platforms such as Google, Facebook, or LinkedIn, ensuring a seamless login experience.

The image shows the project management page for MapMyForest. At the top left is the logo 'MAPMYFOREST'. The top navigation bar includes links for 'Home', 'Projects', 'About', 'Contact', and user icons for profile, notifications, and settings. On the left, there is a sidebar with a 'Projects' section and a 'Recent' link. The main content area displays four project cards: 1. 'Amazon Rainforest Survey' (Creation Date: July 15, 2015, Status: In Progress, orange dot) 2. 'Central Park Tree Census' (Creation Date: July 14, 2008, Status: Completed, blue dot) 3. 'Mountain Forest Assessment' (Creation Date: June 12, 2018, Status: On Hold, red dot) 4. 'Wildlife Migration Study' (Creation Date: April 5, 2012, Status: Completed, blue dot). A search bar labeled 'Search For Project' and a 'See More' link are also present.

Fig 3.6.3: Project Management Page

This page displays a list of current and past projects, including their titles, creation dates, and statuses (e.g., In Progress, Completed) enabling users to efficiently search for and manage their projects.

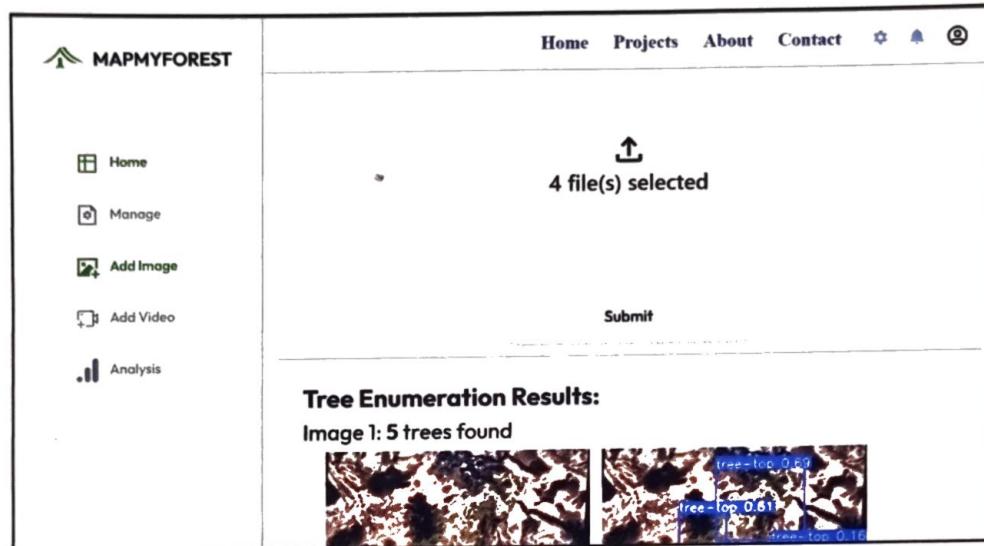


Fig 3.6.4: Image addition Page Layout

This page allows users to upload images or datasets for analysis. Upon submission, the system processes the data and provides results, such as tree enumeration based on the uploaded images, helping users derive insights from their data.

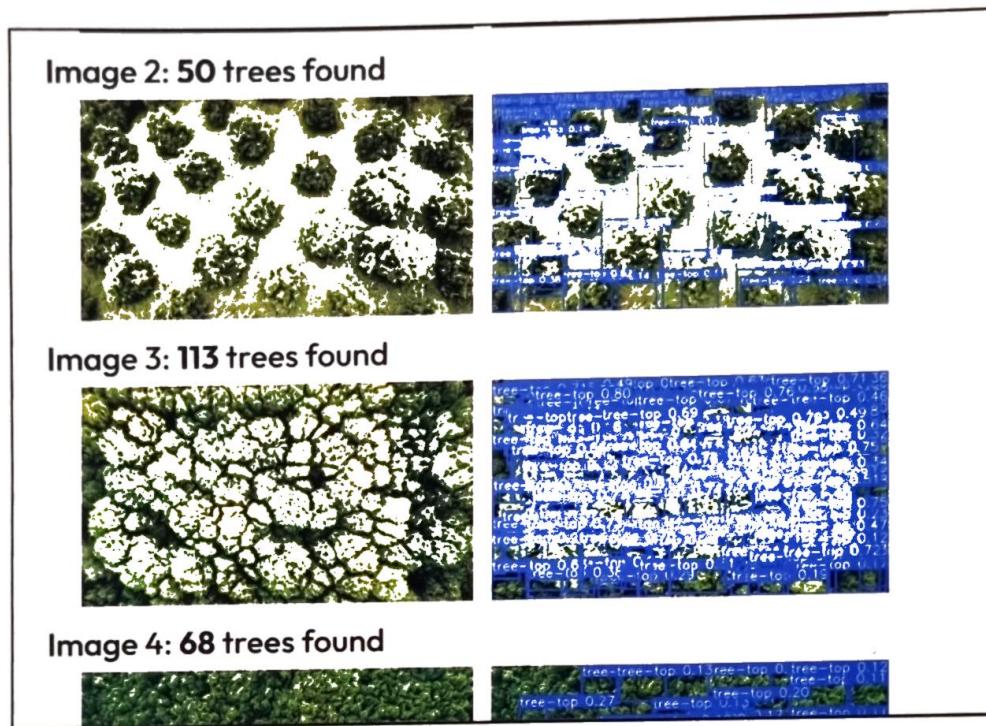


Fig 3.6.5: Enumerated Tree Page

This page shows tree-counting results, with each image displaying the detected trees. For example, image 2 identifies 50 trees, with bounding boxes and confidence scores marking each detection.

Amazon Rainforest Survey

Creation Date: July 15, 2015

Status: In Progress

Selected State: True

Manage Project

Upload Image

Analyze Data

Uploaded Images

Fig 3.6.2: Project Details and Analysis Layout

This page provides an interface for managing a project. Users can perform actions like managing the project, uploading new images, and analyzing data. The project is ongoing, and the current state is active, with uploaded images displayed below.

3.7 Result Analysis and Discussion

Tree Enumeration Model:

The YOLO model gives promising results for diverse scenarios, both for drone and satellite imagery.

One important observation is that the camera's distance from the ground affects the accuracy greatly. To improve this, we can incorporate an option to ask the user for approximate distance from the ground during data acquisition and deciding the patch size for each image based on it, greatly increasing the accuracy regardless of the camera height. Resolution of the image does not affect the accuracy much. This is because the model is trained on very low resolution images helping it detect tree crowns very accurately regardless of resolution.

One more observation is, when the image contains trees on a field with green land or a significant amount of grass, the model might hallucinate and find tree-like patterns in the grass. This can be fixed by clustering along with texture analysis to remove any noise / grass from the image that might be detected as tree crowns.

3.8 Conclusion and Future Work

3.8.1 Conclusion

The "MapMyForest" project successfully demonstrated the feasibility of using satellite imagery and advanced machine learning algorithms to automate tree enumeration and classification. The system significantly enhances the accuracy and efficiency of forest monitoring tasks, reducing the need for manual surveys. The YOLO model, trained for tree detection, proved effective in large-scale forestry applications, while the CNN model facilitated detailed species classification, further enhancing environmental monitoring efforts.

The system is designed for scalability, allowing it to handle large datasets, and it provides real-time analysis capabilities, making it suitable for government agencies, NGOs, and environmental researchers. By integrating these models into a user-friendly software platform, the project offers a robust solution for Environmental Impact Assessments (EIA) and forest management tasks.

3.8.2 Future Work

Several avenues for future development and research have been identified:

- Improved Real-Time Data Processing: Future versions of the system could integrate real-time satellite data feeds for live tree enumeration and classification.
- Enhanced Classification: Incorporating additional environmental data (e.g., weather conditions, soil type) could improve the accuracy of the CNN model in distinguishing between similar species.
- Regional Customization: Developing more region-specific models could address variability in tree types and environmental conditions, further enhancing accuracy in different forest regions.
- 3D Mapping: Future work could also explore integrating 3D models of forests based on aerial imagery, providing better insights into tree height, density, and overall forest structure.

4. References

1. S. Kakade et al., "Satellite Image Analytics for Tree Enumeration for Diversion of Forest Land," SSRN, July 2024.
2. S. Mahule, T. Pranathi, T. Bhuvan Ranjan, T. Asha Kiran, A. Siddhartha, "Intelligent Forest Assessment: Advanced Tree Detection and Enumeration with AI," ISSN 2522-7421, May 2024.
3. M. Brandt et al., "Deep Learning Enables Image-Based Tree Counting, Crown Segmentation, and Height Prediction at National Scale," PNAS Nexus, 2023.
4. S. Mohod et al., "Remote Sensing Application for Analysis of Forest Change Detection," International Conference for Advancement in Technology (ICONAT), 2022.
5. F. Z. Bassine, A. Errami, M. Khaldoun, "Real-Time Algorithm for Tree Detection, Recognition and Counting in a Video Sequence," Journal of Materials and Environmental Science, vol. 11, no. 3, pp. 367-377, 2020.
6. S. Zhu, W. Jing, P. Kang, M. Emam and C. Li, "Data Augmentation and Few-Shot Change Detection in Forest Remote Sensing," in *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 16, pp. 5919-5934, 2023, doi: 10.1109/JSTARS.2023.3285389.
7. Wassim Bouachir, Nizar Bouguila "Computer Vision System for Automatic Counting of Planting Microsites Using UAV Imagery",2019.
8. Y. Wang and Y. Zou, "Fast visual object counting via example-based density estimation," in Proc. IEEE Int. Conf. Image Process. (ICIP), pp. 3653–3657, Sep. 2016.
9. S. V. Koneru et al., "Detection and Enumeration of Trees using Cartosat2 High-Resolution Satellite Imagery," IEEE International Conference on Aerospace Electronics and Remote Sensing Technology (ICARES), Bali, Indonesia, 2018, pp. 1-6, doi: 10.1109/ICARES.2018.8547072, 2018.
10. G. Grenzdörffer, "UAS-based automatic bird count of a common gull colony," Int. Arch. Photogramm., Remote Sens. Spatial Inf. Sci., vol. 1, p. W2, 2013.

**Industry / Inhouse
Research / Innovation:** Inovation

Project Evaluation Sheet 2024-25

Class: D12A/B/C

Title of Project (Group no): Map My Forest (GN - 06)

Mentor Name & Group Members: Dr. Mrs. Gresha Bhattacharya (D12C / 16) Chinmay Deccai (D12C / 53) Shaanveer Singh (D12C / 61) Abhisek Deore (D12C / 15)

	Engineering Concepts & Knowledge	Interpretation of Problem & Analysis	Design / Prototype	Interpretation of Data & Dataset	Modern Tool Usage	Societal Benefit, Safety Consideration	Environment Friendly	Ethics	Team work	Presentation Skills	Applied Engg & Mgmt principles	Life-long learning	Professional Skills	Innovative Appr oach	Total Marks
Review of Project Stage 1	05	04	04	02	04	02	02	02	02	03	02	03	02	05	04
Comments:															

	Engineering Concepts & Knowledge	Interpretation of Problem & Analysis	Design / Prototype	Interpretation of Data & Dataset	Modern Tool Usage	Societal Benefit, Safety Consideration	Environment Friendly	Ethics	Team work	Presentation Skills	Applied Engg & Mgmt principles	Life-long learning	Professional Skills	Innovative Appr oach	Total Marks
Review of Project Stage 1	05	04	04	02	04	02	02	02	02	03	02	03	02	05	04
Comments:															

Name & Signature Reviewer 1
Prerna
Name & Signature Reviewer 2

Date: 23/08/2024