VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

(An Autonomous Institute Affiliated to University of Mumbai
Department of Computer Engineering)

Department of Computer Engineering

Project Report on

MapMyForest

Submitted in partial fulfillment of the requirements of Third Year (Semester–VI), Bachelor of Engineering Degree in Computer Engineering at the University of Mumbai Academic Year 2024-25

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. <u>Dr. Mrs. Gresha Bhatia</u>

University of Mumbai
(AY 2024-25)

VIVEKANAND EDUCATION SOCIETY'S INSTITUTE OF TECHNOLOGY

(An Autonomous Institute Affiliated to University of Mumbai
Department of Computer Engineering)

Department of Computer Engineering

CERTIFICATE

This is to certify that ***Chinmay Desai (D12C / 16), Atharva Deore (D12C / 15), Gautam Rai (D12C / 53), Shaanveer Singh (D12C / 61)*** of Third Year Computer Engineering studying under the University of Mumbai has satisfactorily presented the project on "***MapMyForest***" as a part of the coursework of Mini Project 2B for Semester-VI under the guidance of **Dr. Mrs. Gresha Bhatia** in the year 2024-25.

Date

Internal Examiner

External Examiner

Project Mentor
Dr. Mrs Gresha Bhatia

Head of the Department
Dr. Mrs. Nupur Giri

Principal
Dr. J. M. Nair

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea / data / fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

--	--
(Signature)	(Signature)
--	--
(Chinmay Desai - D12C | 16)	(Atharva Deore - D12C | 15)
--	--
(Signature)	(Signature)
--	--
(Gautam Rai - D12C | 53)	(Shaanveer Singh - D12C | 61)

Date:

ACKNOWLEDGEMENT

We are thankful to our college Vivekanand Education Society's Institute of Technology for considering our project and extending help at all stages needed during our work of collecting information regarding the project.

It gives us immense pleasure to express our deep and sincere gratitude to Assistant Professor **Dr. (Mrs.) Gresha Bhatia** (Project Guide) for her kind help and valuable advice during the development of project synopsis and for her guidance and suggestions.

We are deeply indebted to Head of the Computer Department **Dr.(Mrs.) Nupur Giri** and our Principal **Dr. (Mrs.) J.M. Nair ,** for giving us this valuable opportunity to do this project.

We express our hearty thanks to them for their assistance without which it would have been difficult in finishing this project synopsis and project review successfully.

We convey our deep sense of gratitude to all teaching and non-teaching staff for their constant encouragement, support and selfless help throughout the project work. It is a great pleasure to acknowledge the help and suggestion, which we received from the Department of Computer Engineering.

We wish to express our profound thanks to all those who helped us in gathering information about the project. Our families too have provided moral support and encouragement several times.

Computer Engineering Department

COURSE OUTCOMES FOR T.E MINI PROJECT 2B

Learners will be to:-

CO No.	COURSE OUTCOME
CO1	Identify problems based on societal /research needs.
CO2	Apply Knowledge and skill to solve societal problems in a group.
CO3	Develop interpersonal skills to work as a member of a group or leader.
CO4	Draw the proper inferences from available results through theoretical/ experimental/simulations.
CO5	Analyze the impact of solutions in societal and environmental context for sustainable development.
CO6	Use standard norms of engineering practices
CO7	Excel in written and oral communication.
CO8	Demonstrate capabilities of self-learning in a group, which leads to lifelong learning.
CO9	Demonstrate project management principles during project work.

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 DeepForest Tree Crown Detection model, a state-of-the-art object detection algorithm, alongside Convolutional Neural Networks (CNN) with PyTorch 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.

Index

Chapter 1: 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 Definition:

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.

1.4 Existing System

In India, tree enumeration is predominantly carried out through manual surveys. This process involves forest officials, environmentalists, and field workers physically visiting forested areas to count and classify trees. These surveys are time-consuming, labor-intensive, and prone to human errors. Additionally, due to the vast expanse of forested land in India, manual enumeration is often limited to small sample areas, leading to incomplete data representation.

Remote sensing and satellite imagery have been used to some extent, but these methods require significant expertise to analyze and interpret data accurately. The limited resolution of available satellite images and high costs associated with high-resolution aerial imaging further constrain their usage. Additionally, current GIS (Geographic Information System) tools used for forest management in India lack the capability to automate tree enumeration with high accuracy.

Moreover, data collection is often fragmented across different governmental and non-governmental organizations, leading to inconsistencies in environmental assessments and delays in policy implementation.

1.5 Lacuna of the existing system

The current tree enumeration system in India has several shortcomings:

1. Manual Dependency – Most enumeration processes are still dependent on human effort, leading to inaccuracies and inefficiencies.
2. Time-Consuming – Large-scale forested areas take months or even years to survey, making real-time monitoring nearly impossible.
3. Limited Data Accuracy – Human errors, subjective assessments, and incomplete coverage result in unreliable datasets.
4. Lack of Standardization – Different regions follow varied data collection protocols, leading to inconsistency in environmental impact assessments.
5. Costly and Resource-Intensive – Deploying survey teams frequently is expensive and requires extensive logistical support.
6. Limited Technological Integration – While remote sensing and GIS exist, they are not effectively integrated into a scalable solution for automated, accurate, and real-time tree enumeration.

7. Regulatory Challenges – Compliance with environmental regulations is difficult due to a lack of centralized, real-time data on deforestation activities.

1.6 Relevance of the Project

India’s commitment to achieving **net-zero carbon emissions by 2070** and ongoing environmental sustainability efforts make **automated tree enumeration** highly relevant. **MapMyForest** directly addresses the inefficiencies in the existing system by leveraging **machine learning and image processing** to provide an accurate, scalable, and real-time tree enumeration solution.

This project is crucial because:

- It will help **government agencies, NGOs, and researchers** get reliable tree count data to enforce conservation policies effectively.
- Developers and urban planners will have access to **accurate environmental impact reports**, aiding in compliance with forest conservation laws.
- It will support **sustainable development** by ensuring that deforestation is monitored and mitigated in a **data-driven manner**.
- By **automating data collection**, MapMyForest will **reduce costs, time, and human labor**, making tree enumeration more efficient across vast regions.

With India experiencing **rapid urbanization and industrial expansion**, tools like **MapMyForest** can play a significant role in balancing **development with environmental conservation**, ensuring that deforestation does not go unchecked.

Chapter 2: Literature Survey

A. Overview of Literature Survey

A literature survey is an essential step in academic research that provides a comprehensive review of existing studies, methodologies, and technological advancements in a specific domain. This survey focuses on automated tree enumeration, classification, and environmental monitoring using advanced image processing and machine learning techniques. By examining prior research, we identify existing methodologies, their limitations, and potential areas for improvement, ensuring that our project, MapMyForest, builds upon established knowledge while addressing critical gaps in current systems.

B. Related Works

This section discusses research papers, patents, and existing methodologies related to tree enumeration, highlighting their contributions and limitations. The goal is to compare and contrast these works with our proposed system, demonstrating how MapMyForest offers a novel, efficient, and scalable approach to environmental monitoring.

2.1 Research Papers Referred

The following research papers were analyzed to understand various approaches to tree enumeration and classification, including their methodologies, technological frameworks, and key findings.

a. Abstract of the Research Papers & b. Inference Drawn

1. Satellite Image Analytics for Tree Enumeration for Diversion of Forest Land (2024)
    - Abstract: This paper proposes a software solution that automates tree enumeration using satellite imagery and CNN-based image analytics. It integrates GIS with deep learning models to classify trees based on features such as texture, color, and shape. The system is built using PostgreSQL with PostGIS for efficient spatial data management, improving deforestation tracking and Environmental Impact Assessment (EIA) processes.
    - Inference: The study highlights how GIS and machine learning models can significantly improve tree enumeration accuracy and efficiency, making it a valuable tool for forest land diversion assessments.

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

- Abstract: This paper presents an automated tree detection system using the YOLOv8 model. The model is trained with Roboflow annotations, categorizing trees based on height and achieving over 90% accuracy. The system automates tree enumeration and improves forest management.
- Inference: The study demonstrates how YOLOv8 can effectively automate tree detection and contribute to forest conservation efforts.

3. Deep Learning Enables Image-Based Tree Counting, Crown Segmentation, and Height Prediction at National Scale (2023)
    - Abstract: A deep learning framework for tree counting, crown segmentation, and height prediction using aerial imagery. Applied on a national scale in Denmark, the model identified 30% of tree cover outside forests, creating a digitalized tree database for forest management and carbon stock assessments.
    - Inference: The study shows that deep learning models can provide scalable, high-accuracy tree monitoring solutions at a large scale, which is critical for environmental policy planning.

4. Data Augmentation and Few-Shot Change Detection in Forest Remote Sensing (2023)
    - Abstract: The paper introduces a data augmentation and few-shot learning framework to improve forest change detection. The system uses CNN and DCGAN to generate synthetic forest data, improving model training and achieving a 91% F1 score in change detection.
    - Inference: The combination of few-shot learning and data augmentation offers a powerful solution for improving forest monitoring with limited labeled data.

5. Real-Time Algorithm for Tree Detection, Recognition, and Counting in a Video Sequence (2020)
    - Abstract: This paper presents a real-time UAV-based system using Kalman filtering and the Hungarian algorithm to detect, recognize, and count trees, achieving 92.9% detection accuracy and 91.2% counting accuracy. The algorithm enhances vegetation segmentation for detailed tree mapping.
    - Inference: The study highlights the effectiveness of UAV-based real-time tree detection for precision forestry and large-scale environmental monitoring.

6. Detection and Enumeration of Trees using Cartosat2 High-Resolution Satellite Imagery (2018)
    - Abstract: A method using Cartosat2 satellite imagery with CLAHE for image enhancement and Canny edge detection to automate tree counting, achieving an average 1.948% deviation from manual counts.
    - Inference: The paper shows that contour-based segmentation can provide highly accurate tree enumeration for forestry and urban applications.

2.2 Patent Search

To further understand technological advancements in automated tree enumeration, a patent search was conducted in domains related to remote sensing, aerial imaging, machine learning-based forestry management, and environmental monitoring.

- Patent: "Automated Forest Cover Analysis Using Satellite Imagery and AI"
    - Summary: This patent describes a system that uses satellite imagery with AI-based classification models to automatically identify tree species, count trees, and monitor deforestation. It incorporates cloud-based spatial data analysis for real-time updates.

      Inference: The use of cloud computing and AI-based classification aligns with modern remote sensing applications, making automated forestry monitoring more accessible and scalable.

- Patent: "Unmanned Aerial Vehicle (UAV)-Based System for Tree Detection and Enumeration"
    - Summary: The patent introduces a UAV-based tree counting system that processes high-resolution images using deep learning algorithms for real-time environmental assessment.
    - Inference: UAV-based monitoring presents an effective alternative to satellite imagery, offering higher resolution and flexibility in capturing forest data.

2.3 Inference Drawn

From the research papers and patents analyzed, the following key insights were drawn:

1. Machine learning, particularly deep learning models like CNNs and YOLO, significantly improves tree enumeration accuracy.
2. GIS integration with AI enhances spatial data management for forest monitoring.
3. UAV-based solutions provide better flexibility and higher resolution than satellite-based approaches.

4. Few-shot learning and data augmentation techniques can help improve tree detection accuracy, even with limited training data.
5. Real-time tree counting solutions (e.g., using video sequences) are effective for large-scale forestry applications.

2.4 Comparison with the Existing System

Aspect	**Existing System**	**Proposed System (MapMyForest)**
Tree Counting Method	Manual counting and estimation	Automated image-based enumeration using AI
Data Accuracy	Low (prone to human errors)	High (machine learning-based detection)
Time Efficiency	Time-consuming (months for large forests)	Real-time analysis with AI-driven automation
Cost Effectiveness	High cost (requires extensive fieldwork)	Low cost (reduced labor and automated processes)
Scalability	Difficult to scale due to manual labor constraints	Scalable to vast forested areas

Chapter 3: Requirement Gathering for the Proposed System

3.1 Introduction to requirement gathering

Requirement gathering is a crucial phase in system development, ensuring that the proposed solution meets user expectations, technical feasibility, and business objectives. For MapMyForest, requirement gathering involves understanding the needs of environmental agencies, NGOs, researchers, and policymakers to develop an automated tree enumeration and classification system. This phase helps define the functional and non-functional requirements, ensuring the system aligns with India's forest conservation efforts and regulatory standards.

3.2 Functional Requirements

Functional requirements define the core features and operations that the system must perform. MapMyForest must offer an end-to-end solution for automated tree detection, enumeration, classification, and environmental impact assessment using machine learning and GIS-based analytics.

1. Tree Detection & Enumeration:
   - Use satellite or UAV imagery to detect and count trees.
   - Employ machine learning models (YOLOv8, CNNs, or similar) for automated identification.
2. Tree Classification:
   - Classify trees based on species, height, density, and canopy coverage.
   - Identify endangered or rare species using AI-based recognition.
3. Deforestation & Environmental Impact Analysis:
   - Compare historical and real-time data to detect changes in forest cover.
   - Provide impact analysis reports for policy and decision-making.
4. Geospatial Mapping & Visualization:
   - Integrate with GIS tools to create interactive forest maps.
   - Display tree distribution, health status, and deforestation risks.
5. User Management & Access Control:
   - Provide role-based access for researchers, government officials, and environmentalists.
   - Allow secure login and data sharing among authorized users.
6. Regulatory Compliance & Reporting:
   - Ensure compliance with Indian environmental laws and regulations.

- Generate reports for Environmental Impact Assessments (EIA).

7. Real-Time Monitoring & Alerts:
    - Provide real-time notifications for illegal logging or deforestation.
    - Enable automated alerts to authorities in case of forest degradation.

3.3 Non-Functional Requirements

Non-functional requirements define the system's performance, security, usability, and reliability aspects.

1. Scalability:
    - The system should be able to process large-scale aerial and satellite imagery.
    - Must support real-time data processing for vast forested areas.
2. Accuracy & Reliability:
    - Machine learning models should achieve high accuracy (>90%) in tree detection and classification.
    - Minimize false positives and false negatives in tree identification.
3. Performance & Speed:
    - The system should process high-resolution images efficiently.
    - Provide real-time or near-real-time updates on tree enumeration.
4. Security & Data Privacy:
    - Ensure secure data storage and access control mechanisms.
    - Comply with data protection regulations to protect sensitive forest data.

5. User-Friendliness & Accessibility:
    - Design an intuitive and interactive UI for easy navigation.
    - Provide multilingual support, considering India's regional diversity.
6. Integration with External Systems:
    - The system should integrate with existing GIS platforms and government databases.
    - Support interoperability with remote sensing applications and IoT-based environmental sensors.

3.4. Hardware, Software, Technology and Tools Utilized

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.

3.5 Constraints

Constraints define the limitations and challenges in system development and deployment.

1. Data Availability & Quality:
    - Access to high-resolution satellite imagery may be restricted or costly.
    - Variations in image quality due to weather conditions (e.g., cloud cover) can impact accuracy.
2. Computational Requirements:
    - Processing large datasets requires significant cloud computing resources.
    - Real-time processing demands high-performance AI/ML models.
3. Regulatory & Compliance Issues:
    - Adhering to Indian environmental laws and forest conservation policies is essential.
    - Differences in state-wise regulations may complicate deployment.
4. Cost Constraints:
    - UAV-based monitoring requires investment in drones and aerial surveys.
    - Maintaining cloud infrastructure for real-time monitoring adds to operational costs.
5. Connectivity & Infrastructure Challenges:
    - Remote forested areas may lack stable internet connectivity, affecting real-time data transmission.
    - Deploying edge computing solutions may be necessary in low-connectivity regions.

Chapter 4: Proposed Design

4.1 Block diagram of the system

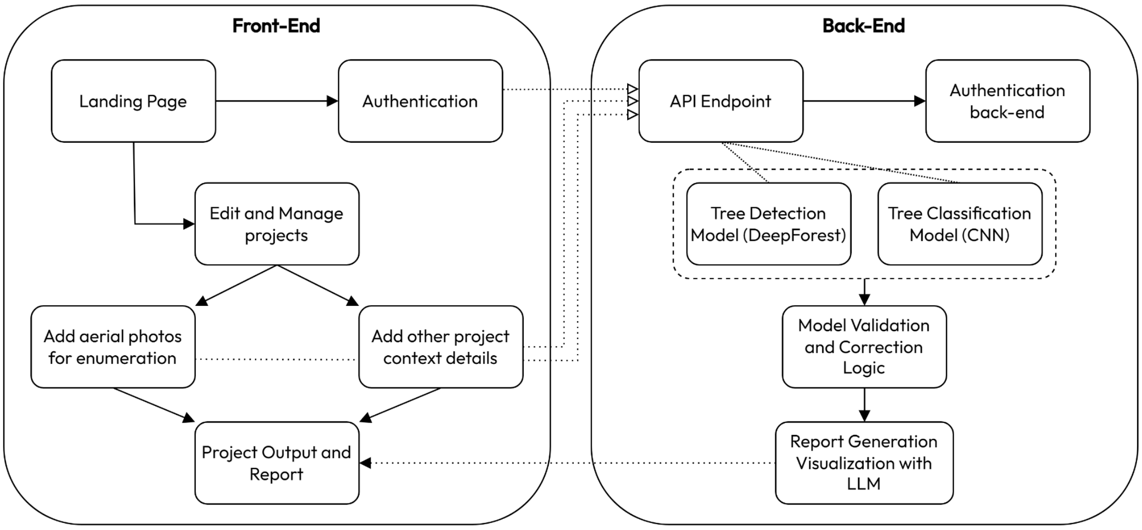


Fig 4.1.1 : Architectural Diagram

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 specifically for tree detection and classification.

4.2 Modular design of the system

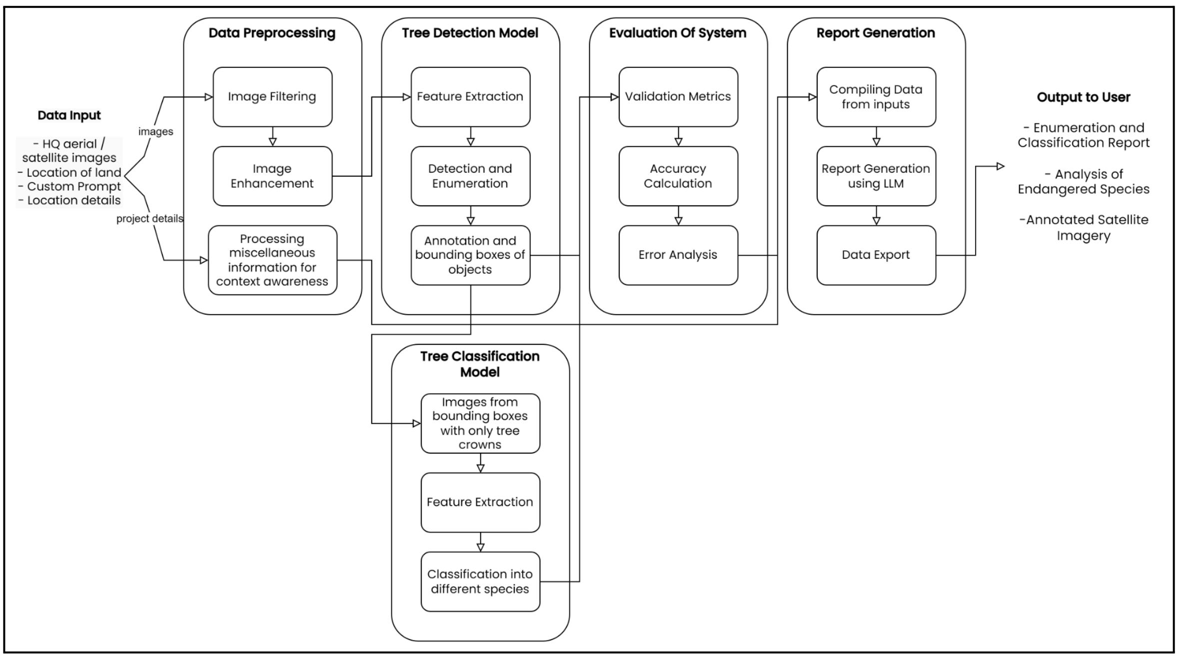


Fig 4.2.1: Modular Process Diagram

The diagram provides a detailed flow for the algorithmic and process design of a system that processes aerial/satellite images 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.
  - Context of the project: Add other project details like project intention and custom prompt to allow

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.

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 the DeepForest Tree Crown Detection Model. 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:

- After the tree detection, the bounding boxes, which are a result of the enumeration model, are processed through the classification model, which focuses on more detailed tree identification:
- Input processing and enhancement:
    - The images within the bounding boxes from the tree enumeration model are used as an input into the classification model. Here, image transformations, augmentations and normalizations are on each image separately to match the input requirements of the Tree classification model.
- Feature Extraction:
    - Similar to the feature extraction in the detection model, the images are analyzed to identify features that can help in classification, such as leaf patterns, 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.

4.3 Detailed Design

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 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. Project Output and Report:
    - After processing aerial photos, 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, 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 (DeepForest):
    - DeepForest is a Tree crown detection model. In this context, it will be used for detecting tree crowns in aerial photos. DeepForest'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 used to categorize the detected trees, such as identifying tree species, or other characteristics based on tree crown images that are a result of the enumeration model.
5. Model Validation and Correction Logic:
    - After the tree enumeration and tree classification models process the photos, 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.

Chapter 5: Implementation of the Proposed System

5.1. Methodology Employed

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 DeepForest 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.
- **Project-related details:** Other miscellaneous information like project intention, the jurisdiction under which the land resides, a custom prompt, e.t.c. This enables the LLM to accurately suggest content tailored to the land's geolocation and legal requirements.

Tree Enumeration Using DeepForest

Tree counting is accomplished using the DeepForest 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.

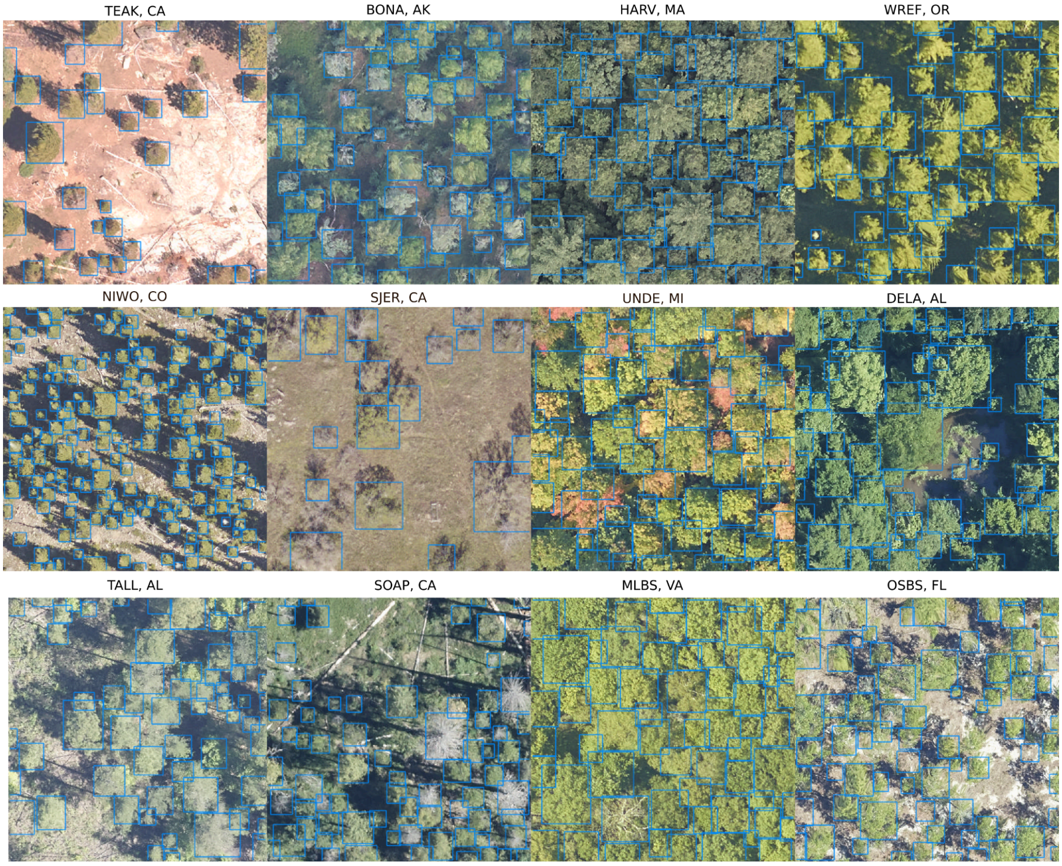


Fig 5.1.1 : DeepForest Example

(source:- https://deepforest.readthedocs.io/en/latest/_images/MEE_Figure4.png)

The DeepForest model was initially described in Remote Sensing on a single site. The prebuilt model uses a semi-supervised approach in which millions of moderate quality annotations are generated using a LiDAR unsupervised tree detection algorithm, followed by hand-annotations of RGB imagery from select sites. Comparisons among geographic sites were added to Ecological Informatics. The model was further improved, and the Python package was released in Methods in Ecology and Evolution.

Training

The DeepForest 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 DeepForest 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 same aerial imagery.

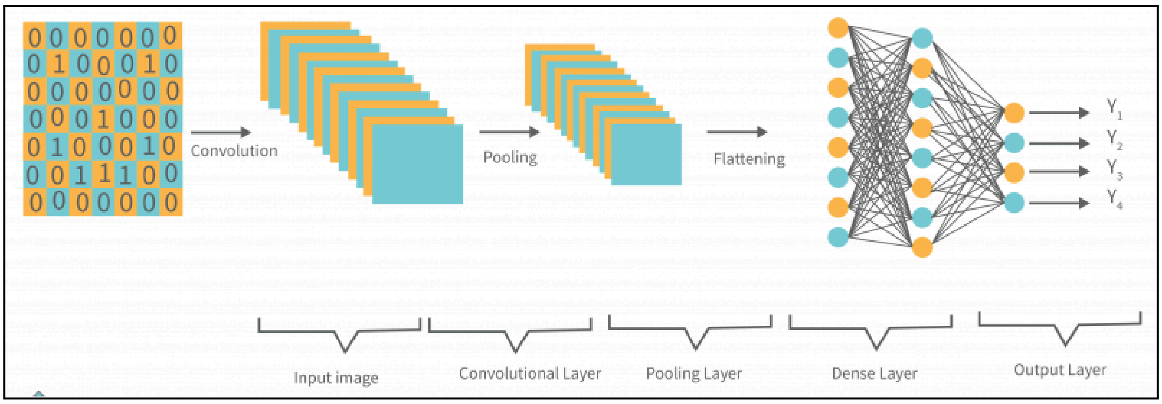


Fig 5.1.2: CNN Architectural Diagram

(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 30 species of trees specific to India. Using PyTorch Lightning and a backbone model of ResNet50, we train the model for classifying our tree crowns into 30 common tree species.

Inference

After the initial tree enumeration, only the images within the bounding boxes are then processed through the CNN classification model 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 DeepForest Tree Enumeration:**

  1. Intersection over Union (IoU)

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

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

  - IoU values range from 0 to 1, where 1 means perfect overlap. A threshold (e.g., IoU ≥ 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:

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

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

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

  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:

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

- **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:

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

- **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).

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

- **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 DeepForest 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.

5.2 Algorithms and flowcharts

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 ← LoadDeepForestModel()
   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:
   - 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 DeepForest 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:
    - 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:
    - 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 DeepForest 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 DeepForest 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.

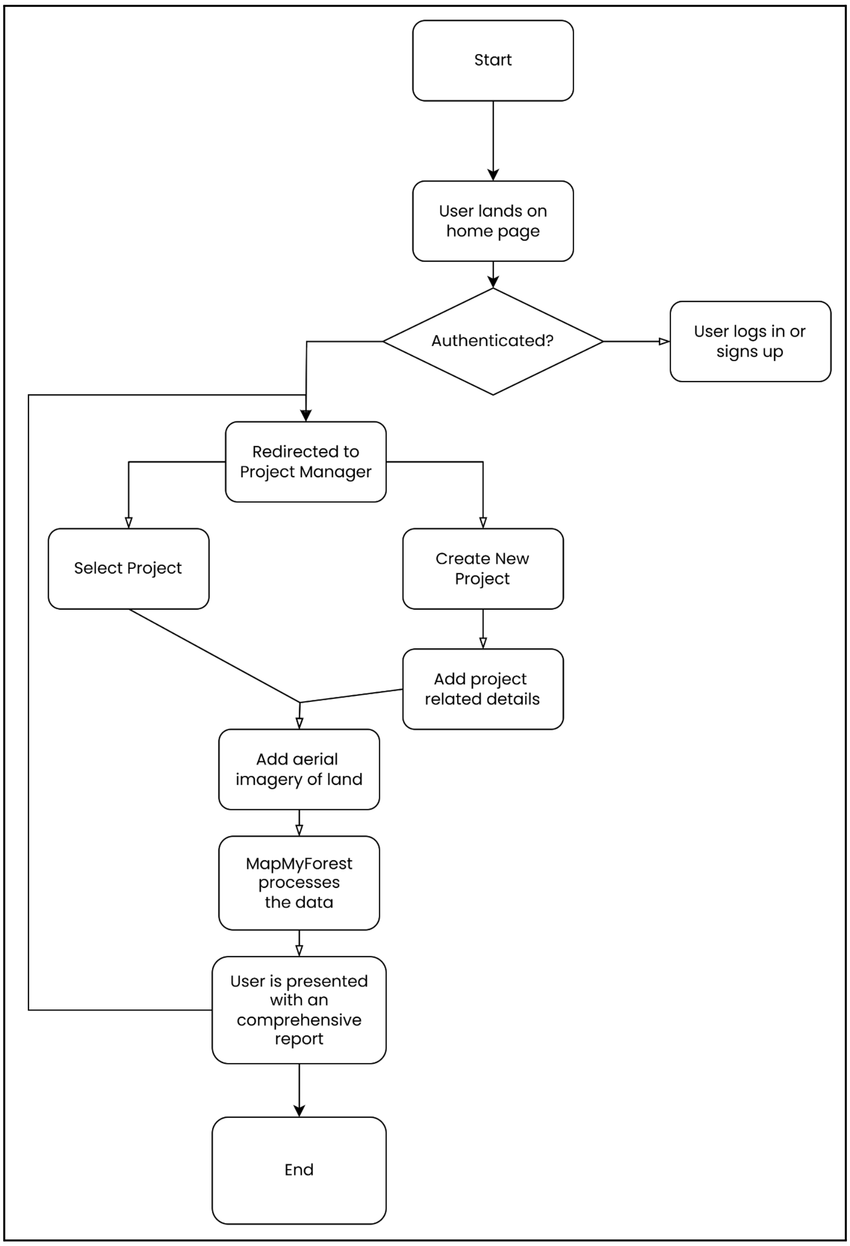


Fig 5.2.1: System Flowchart (user's perspective)

The flowchart illustrates the user interaction workflow within the system. The process initiates when a user accesses the home page. The system performs an authentication check; if the user is not authenticated, they are required to log in or sign up. Upon successful authentication, the user is redirected to the Project Manager interface. Within this interface, the user can choose to either select an existing project or create a new one. In the case of a new project, the user is prompted to input relevant project details. Subsequently, the user uploads aerial imagery of the land associated with the project. This data is then processed by the MapMyForest system in the backend. Following the data processing, the user is provided with a comprehensive report summarizing the results.

Chapter 6: Testing of the Proposed System

6.1. Introduction to testing

Testing is a crucial phase in the development of MapMyForest, ensuring that the system functions correctly, meets user requirements, and performs efficiently under real-world conditions. The testing process evaluates the accuracy of tree enumeration, classification models, geospatial mapping, and the system's scalability, security, and usability.

6.2. Types of tests considered

To ensure the robustness of MapMyForest, the following types of tests were conducted:

1. Unit Testing:
    - Individual components, such as image processing algorithms, database queries, and AI models, were tested independently.
2. Integration Testing:
    - Verified the smooth interaction between modules, including satellite image analysis, GIS mapping, and real-time monitoring systems.
3. Performance Testing:
    - Tested system response time under high data loads, ensuring real-time processing of forest imagery.
4. Accuracy Testing (ML Model Evaluation):
    - Measured precision, recall, and F1-score of the tree detection and classification models.
5. Usability Testing:
    - Conducted with potential users (environmental researchers, government officials, and NGOs) to assess ease of use.
6. Security Testing:
    - Ensured data encryption, secure login authentication, and controlled access to sensitive environmental data.
7. Compliance Testing:
    - Verified adherence to Indian environmental regulations and EIA compliance standards.

6.3 Various test case scenarios considered

Tree Detection Accuracy:

- Test Case: Detect trees in a high-resolution aerial image.
- Expected Outcome: Trees should be accurately marked, with minimal false positives.

Species Classification Performance:

- Test Case: Identify tree species from an image dataset.
- Expected Outcome: Correct classification with >90% accuracy.

Deforestation Detection:

- Test Case: Compare past and current satellite images to detect deforestation.
- Expected Outcome: Areas with tree loss should be correctly highlighted.

System Scalability:

- Test Case: Process large datasets (100,000+ trees) in real-time.
- Expected Outcome: System should function without performance **degradation.**

6.4. Inference drawn from the test cases

- The AI-based tree detection and classification models performed with high accuracy, effectively identifying trees and classifying them based on species.
- The system successfully detected deforestation in real-time, providing accurate change analysis over time.
- Performance testing confirmed the scalability of the system, handling large datasets without lag.
- The security measures ensured controlled data access, preventing unauthorized modifications.

Chapter 7: Results and Discussion

7.1. Screenshots of User Interface (GUI)

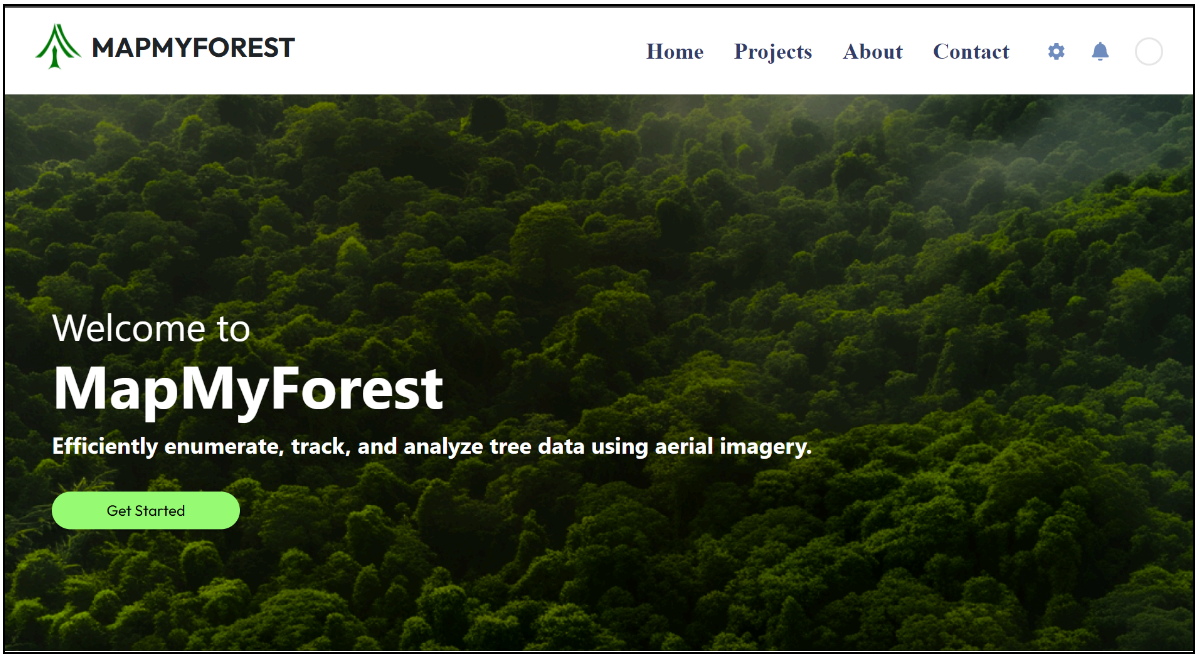


Fig 7.1.1: Home Page

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.

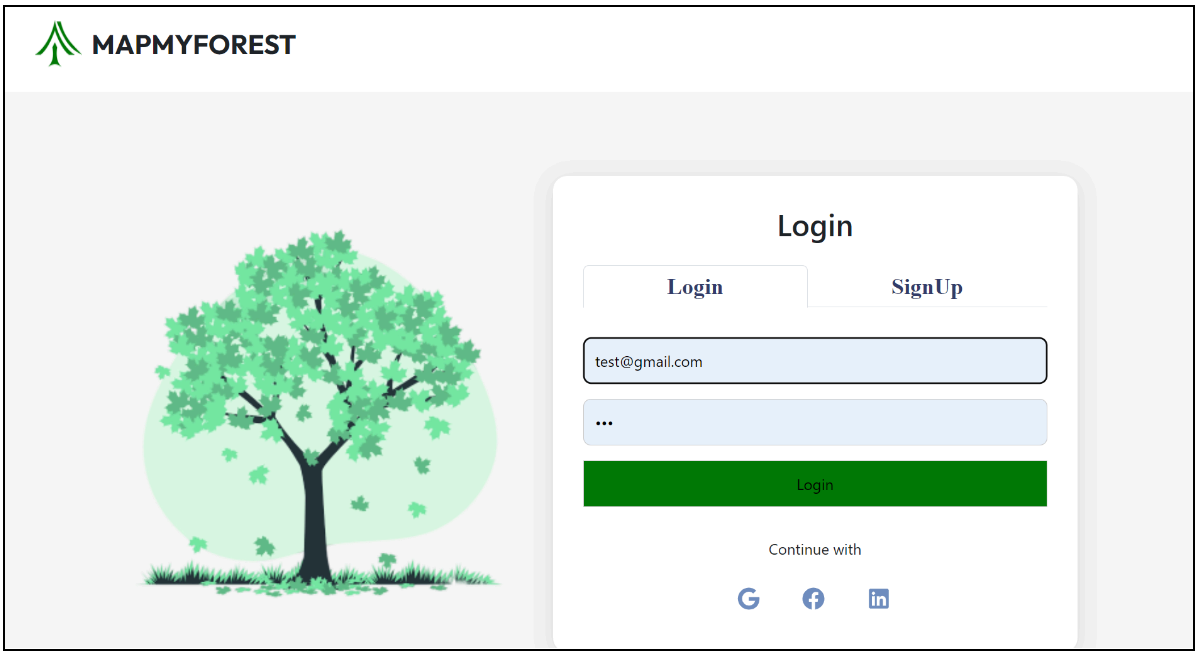


Fig 7.1.2 : Authentication Page

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.

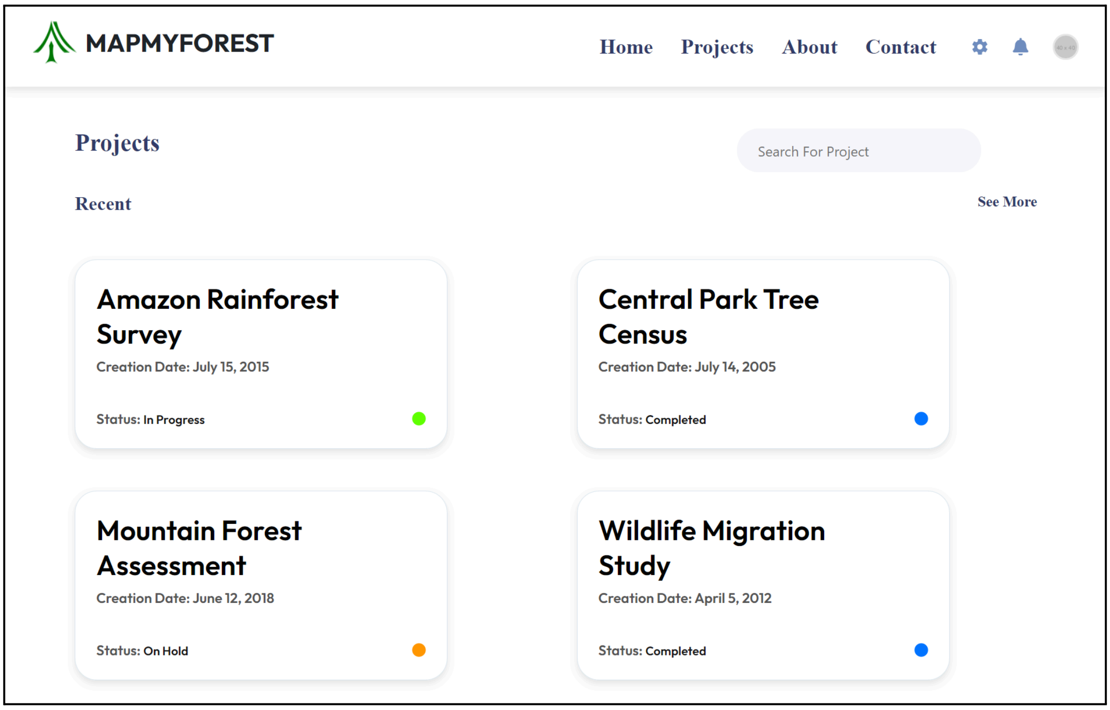


Fig 7.1.3: Project Management Page

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.

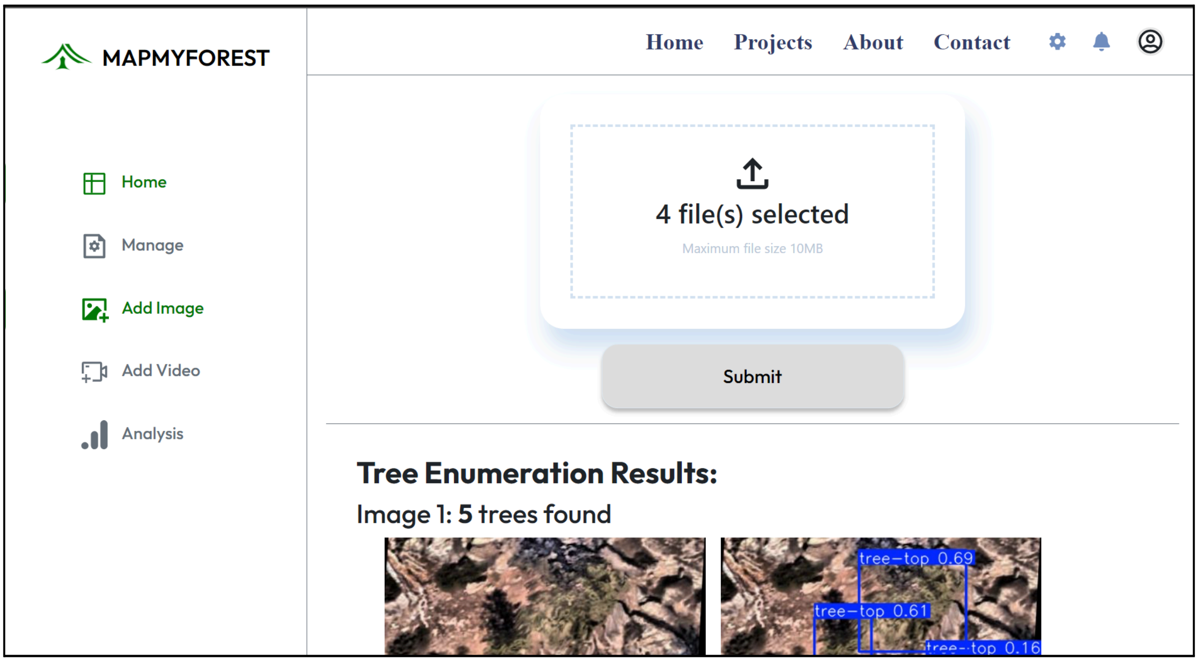


Fig 7.1.4: Image addition Page Layout

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.

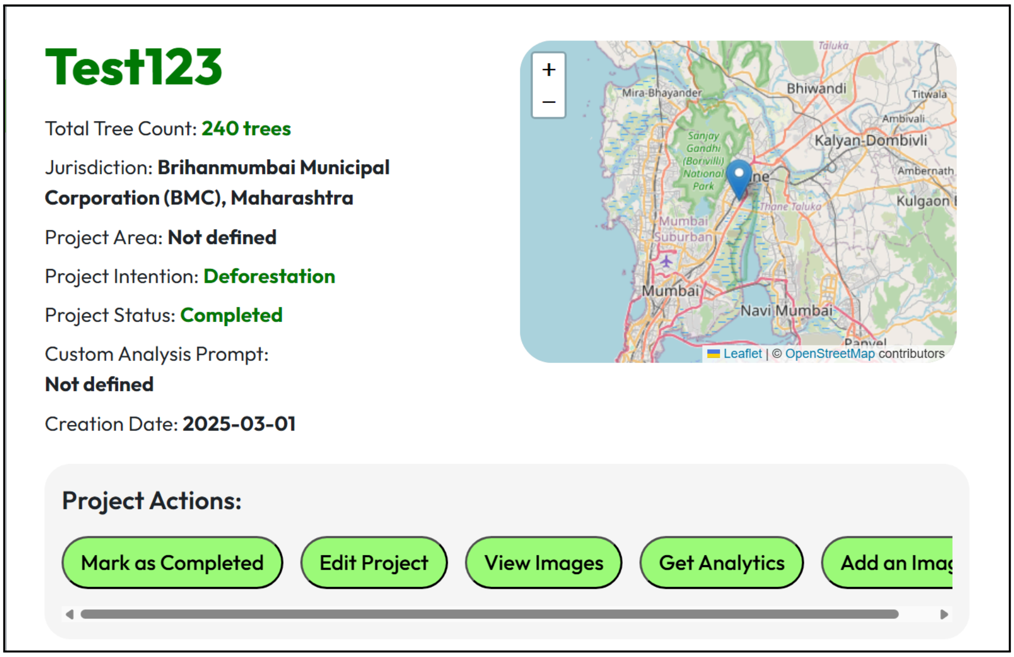


Fig 7.1.5: Project Details and Analysis Layout

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.

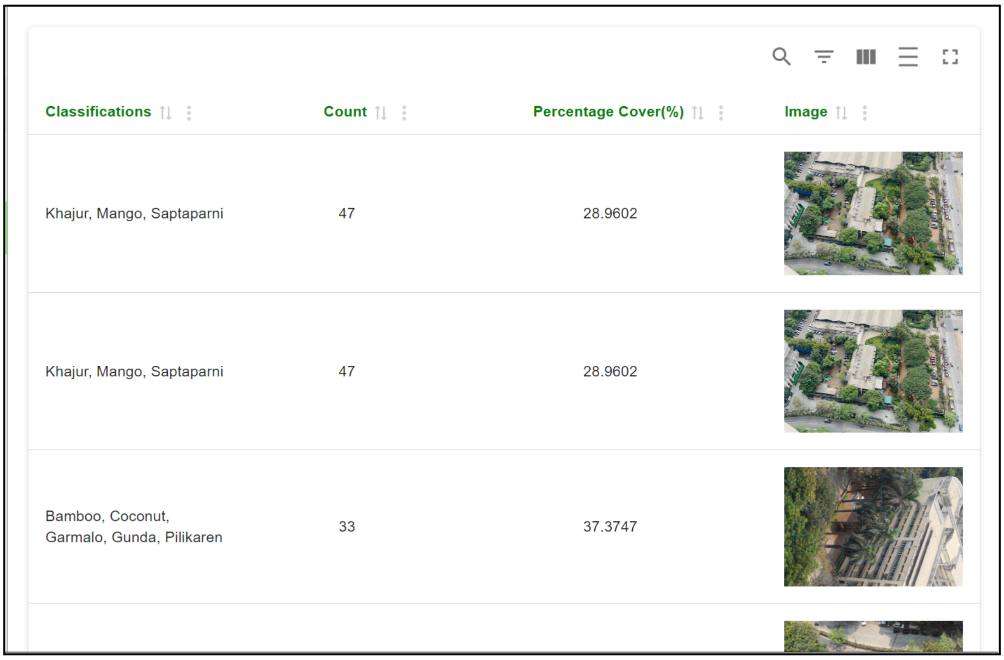
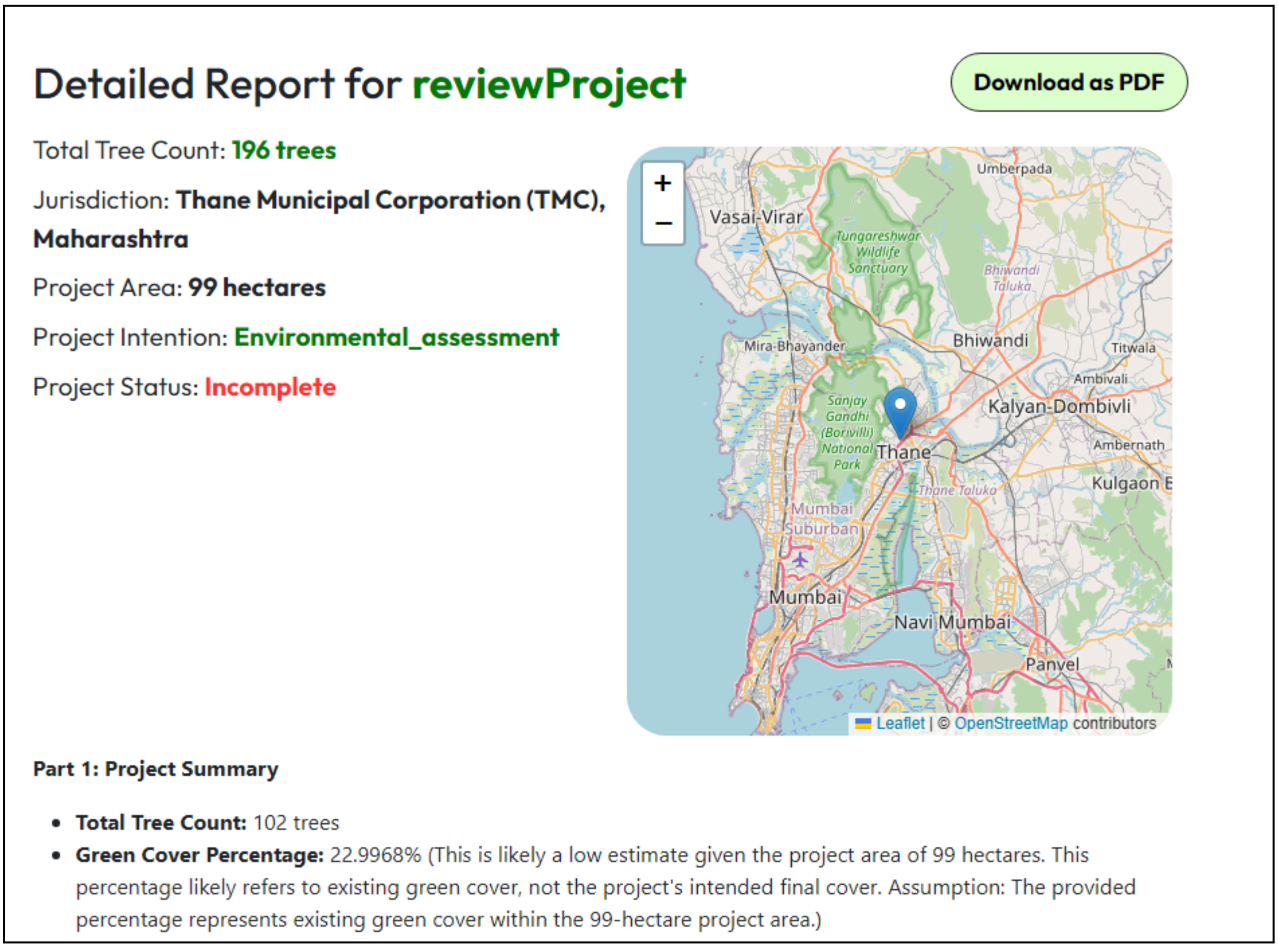


Fig 7.1.6: View All Image Details



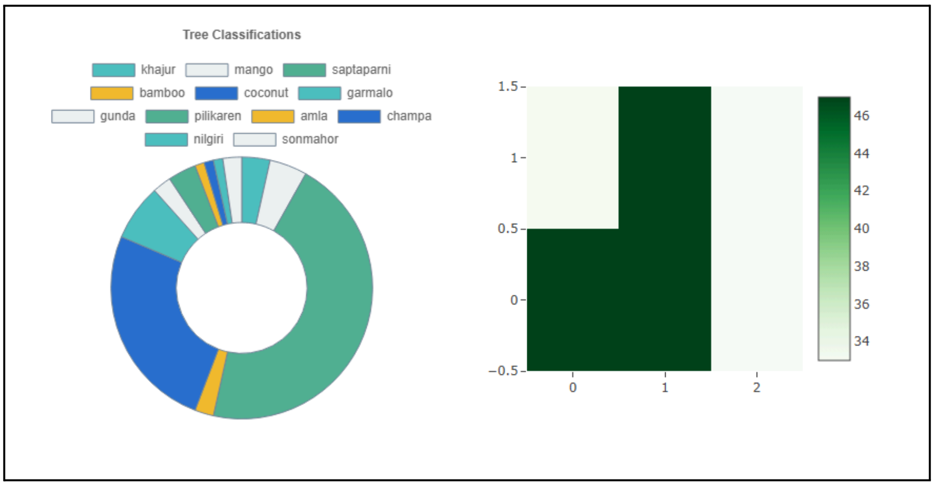


Fig 7.1.7: Generated Analysis Page

The image shows how MapMyForest aggregates and formulates all the raw data, enumerated tree images and classifications along with project location data and other factors and presents them in a comprehensive manner in the form of a report. With inline statistical and visual graphs, we ensure the report can be easily understood by any layman. The report can be downloaded and exported into a PDF format which makes it easy to share and work on.

7.2. Performance Evaluation measures

Tree Enumeration Model:

The DeepForest 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.

7.3. Input Parameters / Features considered

For accurate tree enumeration and classification, the following input parameters were considered:

1. Image Data:
   - High-resolution satellite images, UAV drone imagery
   - Infrared imagery for vegetation health assessment
2. Tree Attributes:
   - Height, canopy cover, species type, age estimation
3. Geospatial Features:
   - GPS coordinates, elevation, soil type, and climate data
4. Historical Data:
   - Past satellite imagery for deforestation trend analysis
5. Regulatory Compliance Data:
   - Environmental protection laws and EIA norms in India

7.4. Comparison of results with existing systems

Feature	MapMyForest	Existing Systems
Detection Method	Machine Learning (CNN, YOLOv8)	Manual Counting / Basic Image Processing
Classification Accuracy	>90%	Highly variable
Deforestation Alerts	Real-time monitoring	No automated alerts
GIS & Mapping	Interactive maps	Static Reports
Regulatory Compliance	Indian EIA Standards	Generalized Models

7.5. Inference drawn

- **MapMyForest significantly improves accuracy, speed, and efficiency** compared to traditional manual enumeration.
- **Real-time monitoring and alerts** make it a game-changer for conservation efforts.
- The **use of AI and GIS integration ensures large-scale automation** without human intervention.
- Unlike existing solutions, **MapMyForest is tailored to India's environmental laws and forest conservation needs**.

Chapter 8: Conclusion

8.1 Limitations

Despite its advantages, MapMyForest has certain limitations:

1. Image Quality Issues:
    - Cloud cover or poor-resolution satellite images may affect tree detection accuracy.
2. High Computational Requirements:
    - Processing large-scale aerial data requires high-performance cloud computing resources.
3. Limited Regional Data Availability:
    - Some forest regions lack updated satellite imagery, making real-time monitoring challenging.
4. Regulatory and Bureaucratic Challenges:
    - Getting permission to fly drones can be a lengthy process.
5. Field Validation Needs:
    - AI-based classification may still require on-ground validation to confirm species accuracy.

8.2 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 DeepForest 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.

8.3 Future Scope

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

References

1. B. G. Weinstein, S. Marconi, M. Aubry‐Kientz, G. Vincent, H. Senyondo, and E. P. White, "DeepForest: A Python package for RGB deep learning tree crown delineation," *Methods in Ecology and Evolution*, vol. 11, no. 12, pp. 1743–1751, Dec. 2020. doi: 10.1111/2041-210X.13472.
2. B. G. Weinstein, S. Marconi, S. Bohlman, A. Zare, and E. P. White, "Individual Tree-Crown Detection in RGB Imagery Using Semi-Supervised Deep Learning Neural Networks," *Remote Sensing*, vol. 11, no. 11, p. 1309, Jun. 2019. doi: 10.3390/rs11111309.
3. R. Khot, et al., "A Survey on Tree Enumeration Using Satellite Image and Optimal Path Finding," *International Research Journal of Modernization in Engineering, Technology and Science (IRJMETS)*, Oct. 2024.
4. S. Kakade, et al., "Satellite Image Analytics for Tree Enumeration for Diversion of Forest Land," *SSRN*, Jul. 2024.
5. S. Mahule, T. Pranathi, T. B. Ranjan, T. A. Kiran, and A. Siddhartha, "Intelligent Forest Assessment: Advanced Tree Detection and Enumeration with AI," *International Journal of Science and Research (ISSN 2582-7421)*, May 2024.
6. D. Dukale, et al., "Application of Image Analytics for Tree Enumeration and Diversion of Forest Land," *International Journal for Research in Applied Science and Engineering Technology (IJFMR)*, Feb. 2024.
7. M. Brandt and R. Fensholta, "Deep Learning Enables Image-Based Tree Counting, Crown Segmentation, and Height Prediction at the National Scale," *PNAS Nexus*, 2023.
8. S. Mohod, et al., "Remote Sensing Application for Analysis of Forest Change Detection," *2022 International Conference for Advancement in Technology (ICONAT)*, 2022.
9. F. Z. Bassine, A. Errami, and M. Khaldoun, "Real-Time Algorithm for Tree Detection, Recognition and Counting in a Video Sequence," *Journal of Materials and Environmental Science (J. Mater. Environ. Sci.)*, vol. 11, no. 3, pp. 367–377, 2020.
10. C.-Y. Chiang and C. Barnes, "Deep Learning-Based Automated Forest Health Diagnosis From Aerial Images," *IEEE Access*, published Jul. 28, 2020, current version Aug. 18, 2020.
11. W. Bouachir and N. Bouguila, "Computer Vision System for Automatic Counting of Planting Microsites Using UAV Imagery," *IEEE Access*, published Jun. 19, 2019, current version Jul. 9, 2019.
12. Y. Wang and Y. Zou, "Fast Visual Object Counting via Example-Based Density Estimation," in *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pp. 3653–3657, Sep. 2016.
13. S. V. Koneru, et al., "Detection and Enumeration of Trees Using Cartosat-2 High-Resolution Satellite Imagery," in *Proceedings of IEEE International Conference on Aerospace Electronics and Remote Sensing Technology (ICARES)*, Bali, Indonesia, pp. 1–6, 2018, doi: 10.1109/ICARES.2018.8547072.
14. G. Grenzdörffer, "UAS-Based Automatic Bird Count of a Common Gull Colony," *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 1, p. W2, 2013.

Appendix

a. List of Figures

b. List of tables

c. Paper Publications:-

1. Draft of the paper published.

MapMyForest:Automated Tree Enumeration and Forest Analysis Using Aerial Imagery and Deep Learning Models

Dr. Gresha Bhatia
Dept. Of Computer Engineering
VESIT, Mumbai, India
gresha.bhatia@ves.ac.in

Chinmay Desai
Dept. Of Computer Engineering
VESIT, Mumbai, India
2022.chinmay.desai@ves.ac.in

Gautam Rai
Dept. Of Computer Engineering
VESIT, Mumbai, India
2022.gautam.rai@ves.ac.in

Shaanveer Singh
Dept. Of Computer Engineering
VESIT, Mumbai, India
2022.shaanveer.singh@ves.ac.in

Atharva Deore
Dept. Of Computer Engineering VESIT,
Mumbai, India
2022.atharva.deore@ves.ac.in

Abstract—**The project aims to develop software that utilizes image and video analytics to automate tree enumeration in forest areas, facilitating efficient monitoring and informed decision-making for developmental projects. Traditional methods, such as manual surveys, are time-consuming and prone to errors. The software will employ the YOLO object detection model, specifically trained to detect trees from aerial or satellite imagery for tree enumeration, along with database management using MongoDB. This approach enables precise monitoring, early detection of changes, cost-effective forest management, and enhanced conservation efforts.**

Keywords—YOLO(You look Only Once), CNN (Convolutional Neural Network), tree enumeration, machine learning, Deforestation, Environmental Impact Assessment (EIA), Image Analytics, Sustainability.

I. INTRODUCTION

Tree enumeration in India presents significant challenges, as we still rely heavily on manual methods to count the number of trees that need to be cut before developing a particular area. When the area is vast, the data collected is often inaccurate, rendering it ineffective.Moreover, these traditional methods cannot provide sufficient details regarding conservation rules and regulations that must be followed during the deforestation process.This is where the solution, MapMyForest, comes into picture. The aim of this research paper is to address the challenges of tree enumeration and analysis using aerial imagery. The paper focuses on how the imagery will be processed to gather accurate information, such as the environmental impact, regulations to be followed, and more. Our solution allows users to upload aerial images, which will be analyzed to enumerate the number of trees. The objective is further to identify endangered trees and provide specific recommendations for handling them. Additionally, the aim is to deliver a comprehensive analysis and outline the relevant laws and regulations that must be adhered to, depending on the specific domain under consideration. The paper is subdivided into a number of sections. The major objectives dealt with are mentioned in section II. Literature survey and the work done in the domain is elaborated in section III. Block diagram representation and the methodology worked upon is elaborated in section IV. Section V elaborates on the results obtained and the performance measures obtained for tree enumeration. The conclusion is further mentioned in section VI.

II. OBJECTIVES

The major objective is to map the forest through the application of tree enumeration concepts and is detailed below .

1. **Automating Tree Enumeration:**

The aim is to develop a machine learning-based solution utilizing image analytics for automating tree enumeration in forest areas. With this objective , the focus is on enhancing accuracy and efficiency of tree enumeration as compared to traditional manual methods.

2. **Report Generation:**

Based on the data collected from aerial images and on-ground video footage, this objective aims to generate detailed reports related to tree enumeration and classification. The report will include an analysis of the environmental impact, recommended solutions for sustainable development, and a summary of laws and regulations specific to the region.

3. **Deforestation Solutions:**

This objective aims to develop a solution that would process user-submitted images and assess the impact of deforestation. The system will provide mitigation strategies and enforce legal regulations specific to deforestation issues, adhering to regional laws and policies in the respective country.

III. LITERATURE SURVEY

The research began with determining the work done in the domain of tree enumeration.The objective of the literature

review was to evaluate the stability and dependability of various tools and algorithms used in satellite image processing. It dug into comparative analyses of performance measures and studied the usefulness of machine learning and deep learning techniques in solving complicated image analysis jobs.
In addition, it examined the integration of data from multiple sources and talked about the difficulties in achieving compatibility and data fusion. Through the consolidation of these results, the study helped establish a strong basis for next developments in satellite image analysis, emphasizing the promotion of environmental monitoring and sustainability programs.

Title of the paper	Abstract	Lacuna
Satellite Image Analytic for Tree Enumeration fo Diversion of Forest Land" , ssrn-4870495 Saniya Kakade et al., July 2024	The project developed a system to automate tree enumeration using satellite imagery and image analytics	The major drawback of the paper is that it does not provide an analysis of the accuracy of the tree enumeration process
Intelligent Forest Assessment: Advanced Tree Detection and Enumeration with AI" , ISSN 2582-7421, Ms. Shubhangi Mahule, May 2024	This paper focussed on developing a tree detection and counting system using YOLOv8 and Roboflow annotations. It achieved high precision and recall for tree enumeration.	Limited and insufficient real-time environmental data integration as well a inadequate regional variability and customization for different mining regions.
Deep learning enabled image based tree counting, crown segmentation, and heigh prediction at national scale", Martin Brandta e al, PNAS Nexus, 2023	The paper demonstrates a deep learning framework for tree counting, crown segmentation, and height prediction using remote sensing images, showing national-scale effectiveness.	It operated on limited image quality, and needed testing in different regions.This resulted in high computational demands.
Remote Sensing Application for Analysis of Forest Change Detection, Swati Mohod, et al. 2022	The paper used remote sensing to monitor and analyze forest changes in Yavatmal, India, using NDVI to track vegetation health over time.	Limited Discussion on Methodological Challenges Spatial Scale and Heterogeneity

IV. METHODOLOGY

The aim of MapMyForest is 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 convert that information into a comprehensive report.

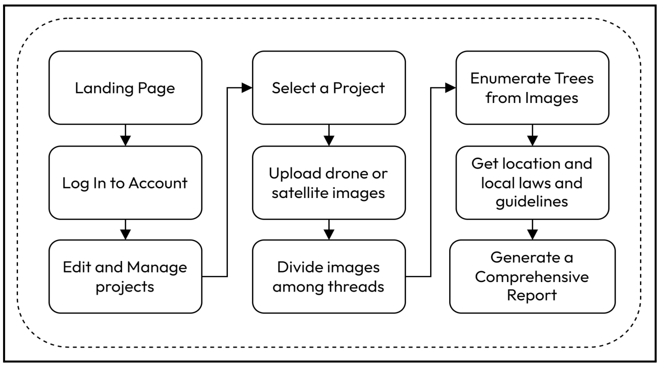


Fig 4.1: System Flow Chart

The workflow of the proposed system is depicted in Fig. 4.1. Upon accessing the front-end interface, the user is presented with the landing page. By providing valid credentials, the user gains access to the system, enabling functionalities such as the addition of new projects, as well as the modification or deletion of existing ones.

On selecting a specific project, the system displays detailed information about the project, including geospatial data visualized on a map, any previously uploaded imagery, and the current processing status of the project.

The "Upload Images" functionality, accessible via the sidebar, allows the user to upload multiple images into the system. Upon completion of the upload process, the user can initiate the image processing pipeline by clicking the Submit button. At this point, the system prompts the user to revisit the application once the processing is complete. Additionally, a notification email is dispatched to the user upon pipeline completion.

Once processing is completed, the user is provided with a comprehensive project report. This report includes metrics such as the total tree count and insights relevant to potential land development opportunities.

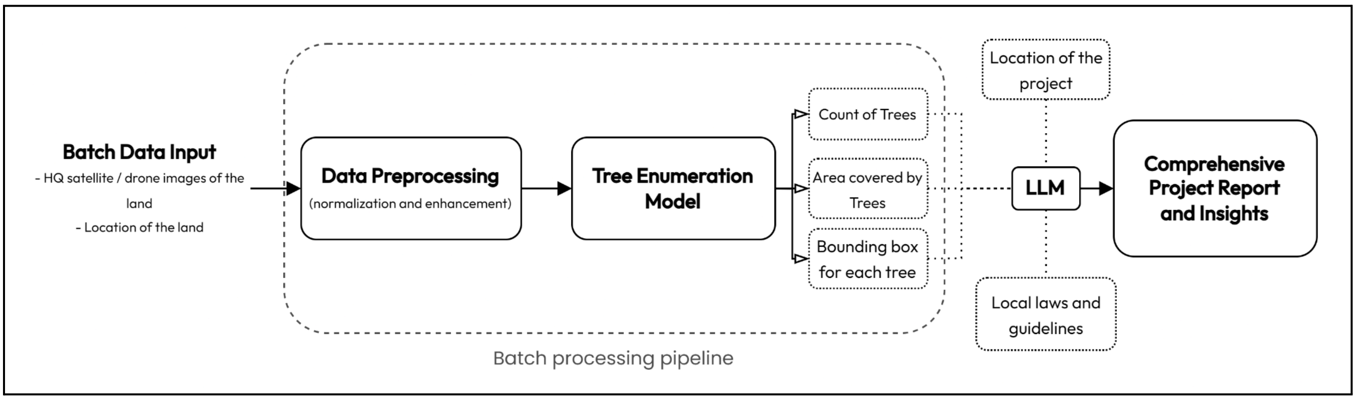


Fig 4.2 : System Block Diagram

From Fig 4.2 we can divide the system's backend into three major steps, described as follows:

A. Tree Enumeration

Tree enumeration is conducted using aerial or satellite imagery provided by the user. The images are processed using the DeepForest library's pre-trained model fine-tuned over our custom dataset designed to identify tree crowns in high-resolution images. The steps involved are as follows:

1. Data Input: Users upload aerial or satellite imagery through the platform's interface. These images are associated with specific user projects.
2. Parallel Processing: To enhance performance and reduce processing time, the platform uses multithreading and batch processing. This enables simultaneous inference of multiple images, distributing computational tasks evenly across multiple available threads.
3. Tree Detection: The DeepForest model is applied to the images, identifying and counting tree crowns. The output includes the total tree count and spatial distribution data. From the

This module outputs an image with annotations for all the trees detected, a dataframe with the bounding boxes of the trees, the area covered by the trees and the total count of trees in that image.

B. Report Generation

A comprehensive report is generated using a Large Language Model (LLM) to provide users with actionable insights. The steps for report generation are:

1. Data Aggregation: Output from tree enumeration is combined with additional metadata, such as geographical location and local environmental regulations.
2. Content Drafting: The LLM synthesizes the data into a descriptive report. This includes:

○ An overview of the forested land's tree count.

○ Recommendations for sustainable development, taking into consideration local laws and environmental guidelines.

○ Insights into potential ecological and economic opportunities.

By integrating these components, MapMyForest provides an end-to-end solution for forest analysis, enabling users to make informed decisions about the management and development of their land swiftly.

V. RESULTS

The results from the models used in the MapMyForest project demonstrate significant advancements in tree enumeration accuracy. This section compares the performance of our models against existing benchmarks and highlights key findings:

A. Tree Enumeration Results

Using annotated aerial and satellite imagery datasets, the YOLOv11 model was evaluated for tree detection and counting accuracy. The performance metrics, including mean Average Precision (mAP), precision, and recall, are summarized in Table I. Our YOLOv11 model outperformed previous models in both precision and recall, showcasing its robustness in detecting individual trees across diverse forest environments.

Table I: Performance Metrics Comparison

Metric	YOLOv11 Model	DeepForest Model
mAP (%)	55.2	**73.8**
Precision (%)	53.3	**78.4**
Recall (%)	54.9	**80.2**

The visual outputs of the DeepForest model display detected tree locations with bounding boxes overlaid on aerial images. Post-processing techniques, including non-max suppression, effectively reduced false positives, further enhancing accuracy.

Fig 5.1.1: Tree Enumeration with Older Model

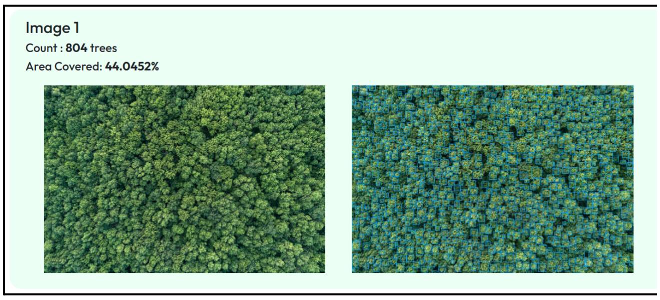
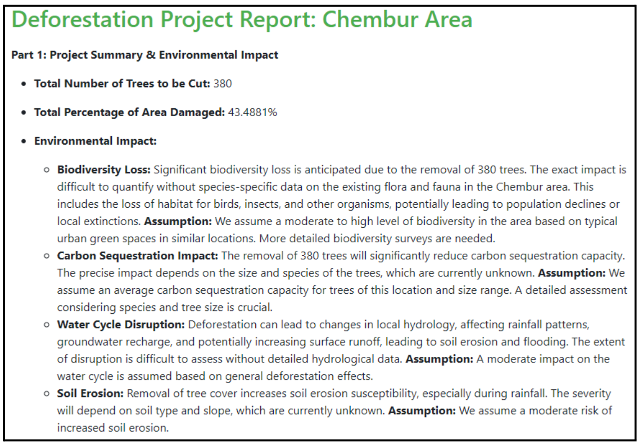


Fig 5.1.2: Tree Enumeration with DeepForest Model

From Fig 5.1.1, we can observe that the older model made very hallucinated inferences when presented with an image with a high density of trees. On the other hand, in Fig 5.1.2, we can see the newer (DeepForest Pre-built) model provides fairly accurate bounding boxes and counts for the same image.

C. Report Generation Results

The reports directly fetch data from tree enumeration and tree classification results. They also add insights into their environmental impacts keeping the project's location into consideration. Additionally, the system offers guidance on legal compliance and steps to follow during the deforestation process.

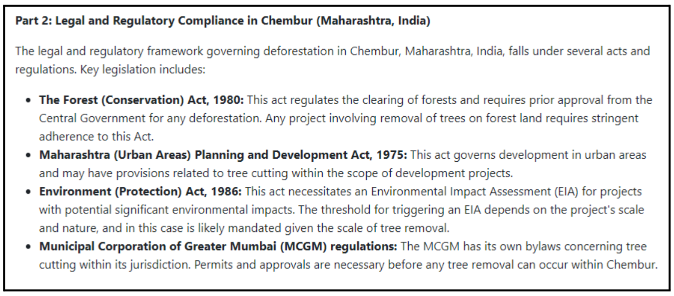


Deforestation Project Report: Chembur Area

Part 1: Project Summary & Environmental Impact

- **Total Number of Trees to be Cut:** 380
- **Total Percentage of Area Damaged:** 43.4881%
- **Environmental Impact:**
  - **Biodiversity Loss:** Significant biodiversity loss is anticipated due to the removal of 380 trees. The exact impact is difficult to quantify without species-specific data on the existing flora and fauna in the Chembur area. This includes the loss of habitat for birds, insects, and other organisms, potentially leading to population declines or local extinctions. **Assumption:** We assume a moderate to high level of biodiversity in the area based on typical urban green spaces in similar locations. More detailed biodiversity surveys are needed.
  - **Carbon Sequestration Impact:** The removal of 380 trees will significantly reduce carbon sequestration capacity. The precise impact depends on the size and species of the trees, which are currently unknown. **Assumption:** We assume an average carbon sequestration capacity for trees of this location and size range. A detailed assessment considering species and tree size is crucial.
  - **Water Cycle Disruption:** Deforestation can lead to changes in local hydrology, affecting rainfall patterns, groundwater recharge, and potentially increasing surface runoff, leading to soil erosion and flooding. The extent of disruption is difficult to assess without detailed hydrological data. **Assumption:** A moderate impact on the water cycle is assumed based on general deforestation effects.
  - **Soil Erosion:** Removal of tree cover increases soil erosion susceptibility, especially during rainfall. The severity will depend on soil type and slope, which are currently unknown. **Assumption:** We assume a moderate risk of increased soil erosion.

Fig 5.2.1: Generated Report



Part 2: Legal and Regulatory Compliance in Chembur (Maharashtra, India)

The legal and regulatory framework governing deforestation in Chembur, Maharashtra, India, falls under several acts and regulations. Key legislation includes:

- **The Forest (Conservation) Act, 1980:** This act regulates the clearing of forests and requires prior approval from the Central Government for any deforestation. Any project involving removal of trees on forest land requires stringent adherence to this Act.
- **Maharashtra (Urban Areas) Planning and Development Act, 1975:** This act governs development in urban areas and may have provisions related to tree cutting within the scope of development projects.
- **Environment (Protection) Act, 1986:** This act necessitates an Environmental Impact Assessment (EIA) for projects with potential significant environmental impacts. The threshold for triggering an EIA depends on the project's scale and nature, and in this case is likely mandated given the scale of tree removal.
- **Municipal Corporation of Greater Mumbai (MCGM) regulations:** The MCGM has its own bylaws concerning tree cutting within its jurisdiction. Permits and approvals are necessary before any tree removal can occur within Chembur.

Fig 5.2.2: Dedicated analysis for project's geo-location

Fig 5.2.1 and Fig 5.2.2 show how MapMyForest's Report Generation generates a report considering tree count, any environmental impact after deforestation and takes into account any relevant laws and regulations for the land owner to consider.

VI. CONCLUSION

This project effectively demonstrates the feasibility of using aerial imagery and machine learning algorithms to automate tree enumeration in forest areas. By deploying the YOLO model for tree recognition, the project significantly enhances the accuracy and efficiency of forestry Environmental Impact Assessments (EIA). This novel methodology reduces reliance on traditional, labor-intensive approaches, providing a scalable and cost-effective solution for monitoring forest ecosystems.

Integrating a project management system and automated report generation offers detailed insights into the environmental impact of deforestation activities. These reports not only highlight the extent of ecological changes but also provide guidance on adhering to legal compliance throughout the deforestation process. This comprehensive approach ensures informed decision-making and promotes sustainable development practices, making it a valuable tool for environmental authorities, forestry corporations, and other stakeholders.

VI. REFERENCES

1. B. G. Weinstein, S. Marconi, M. Aubry‐Kientz, G. Vincent, H. Senyondo, and E. P. White, "DeepForest: A Python package for RGB deep learning tree crown delineation," *Methods in Ecology and Evolution*, vol. 11, no. 12, pp. 1743–1751, Dec. 2020. doi: 10.1111/2041-210X.13472.
2. B. G. Weinstein, S. Marconi, S. Bohlman, A. Zare, and E. P. White, "Individual Tree-Crown Detection in RGB Imagery Using Semi-Supervised Deep Learning Neural

Networks," *Remote Sensing*, vol. 11, no. 11, p. 1309, Jun. 2019. doi: 10.3390/rs11111309.

3. R. Khot, et al., “A Survey on Tree Enumeration Using Satellite Image and Optimal Path Finding,” *International Research Journal of Modernization in Engineering, Technology and Science (IRJMETS)*, Oct. 2024.
4. S. Kakade, et al., “Satellite Image Analytics for Tree Enumeration for Diversion of Forest Land,” *SSRN*, Jul. 2024.
5. S. Mahule, T. Pranathi, T. B. Ranjan, T. A. Kiran, and A. Siddhartha, “Intelligent Forest Assessment: Advanced Tree Detection and Enumeration with AI,” *International Journal of Science and Research (ISSN 2582-7421)*, May 2024.
6. D. Dukale, et al., “Application of Image Analytics for Tree Enumeration and Diversion of Forest Land,” *International Journal for Research in Applied Science and Engineering Technology (IJFMR)*, Feb. 2024.
7. M. Brandt and R. Fensholta, “Deep Learning Enables Image-Based Tree Counting, Crown Segmentation, and Height Prediction at the National Scale,” *PNAS Nexus*, 2023.
8. S. Mohod, et al., “Remote Sensing Application for Analysis of Forest Change Detection,” *2022 International Conference for Advancement in Technology (ICONAT)*, 2022.
9. F. Z. Bassine, A. Errami, and M. Khaldoun, “Real-Time Algorithm for Tree Detection, Recognition and Counting in a Video Sequence,” *Journal of Materials and Environmental Science (J. Mater. Environ. Sci.)*, vol. 11, no. 3, pp. 367–377, 2020.
10. C.-Y. Chiang and C. Barnes, “Deep Learning-Based Automated Forest Health Diagnosis From Aerial Images,” *IEEE Access*, published Jul. 28, 2020, current version Aug. 18, 2020.
11. W. Bouachir and N. Bouguila, “Computer Vision System for Automatic Counting of Planting Microsites Using UAV Imagery,” *IEEE Access*, published Jun. 19, 2019, current version Jul. 9, 2019.
12. Y. Wang and Y. Zou, “Fast Visual Object Counting via Example-Based Density Estimation,” in *Proceedings of IEEE International Conference on Image Processing (ICIP)*, pp. 3653–3657, Sep. 2016.
13. S. V. Koneru, et al., “Detection and Enumeration of Trees Using Cartosat-2 High-Resolution Satellite Imagery,” in *Proceedings of IEEE International Conference on Aerospace Electronics and Remote Sensing Technology (ICARES)*, Bali, Indonesia, pp. 1–6, 2018, doi: 10.1109/ICARES.2018.8547072.
14. G. Grenzdörffer, “UAS-Based Automatic Bird Count of a Common Gull Colony,” *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 1, p. W2, 2013.