Project Report On

DEFORESTATION DETECTION USING SATELLITE IMAGES

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science and Engineering

By

Richard Sherlin(U2003167)

Roy Rajesh(U2003177)

Shaun Tojan (U2003194)

Tijin T Babu (U2003210)

Under the guidance of

Mr. Harikrishnan M

Department of Computer Science and Engineering
Rajagiri School of Engineering & Technology (Autonomous)
(Parent University: APJ Abdul Kalam Technological University)
Rajagiri Valley, Kakkanad, Kochi, 682039
April 2024

CERTIFICATE

This is to certify that the project report entitled ***"DEFORESTATION DETECTION USING SATELLITE IMAGES"*** *is a bonafide record of the work done by* ***Richard Sherlin (U2003167)****,* ***Roy Rajesh (U2003177)****,* ***Shaun Tojan (U2003194)****,* ***Tijin T Babu (U2003210)****submitted to the Rajagiri School of Engineering & Technology (RSET) (Autonomous) in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2023-2024.*

Mr. Harikrishnan M
Project Guide
Assistant Professor
Dept. of CSE
RSET

Ms. Anita John
Project Co-ordinator
Assistant Professor
Dept. of CSE
RSET

Dr. Preetha K.G.
Head Of The Department
Professor
Dept. of CSE
RSET

ACKNOWLEDGMENT

We wish to express our sincere gratitude towards **Dr. Sreejith P S**, Principal of RSET, and **Dr. Preetha K.G.**, Head of the Department of Computer Science And Engineering for providing us with the opportunity to undertake this project, DEFORESTATION DETECTION USING SATELLITE IMAGES.

We are highly indebted to our project coordinators, **Ms. Anita John**, Assistant Professor, Department of Computer Science and Engineering and **Mr. Sajanraj T.D.**, Assistant Professor, Department of Computer Science and Engineering, for their valuable support.

It is indeed our pleasure and a moment of satisfaction for us to express our sincere gratitude to our project guide **Mr. Harikrishnan M** for his patience and all the priceless advice and wisdom he has shared with us.

Last but not the least, We would like to express our sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Richard Sherlin

Roy Rajesh

Shaun Tojan

Tijin T Babu

Abstract

Our final year project focuses on a comprehensive deforestation trend analysis and predictor system. ForestEye is a web application designed to address the pressing issue of deforestation in the state of Para, Brazil. Leveraging machine learning, specifically the Random Forest algorithm, ForestEye provides users with two key functionalities: a deforestation predictor and a trend analyzer.

The deforestation predictor allows users to select a point on a map of Para using latitude and longitude coordinates and choose a future year beyond 2024. Based on historical forest area data from 2008 to 2024, the predictor forecasts the amount of deforestation in square kilometers for the selected area, aiding in understanding potential future environmental impacts.

On the other hand, the trend analyzer enables users to explore deforestation patterns over time. Users can select a year gap and a country (India, Zambia, Indonesia, or Brazil) to visualize deforestation trends as a bar graph. This feature provides valuable insights into the historical progression of deforestation, aiding policymakers, researchers, and conservationists in making informed decisions.

By combining geospatial data, machine learning, and interactive visualization, ForestEye empowers stakeholders with the tools needed to monitor, analyze, and predict deforestation trends in Para. Its user-friendly interface and powerful analytical capabilities make it a valuable asset in the ongoing efforts to protect and preserve the world's forests.

Contents

Acknowledgment i

Abstract ii

List of Abbreviations vi

List of Figures viii

List of Tables ix

1 Introduction 1

1.1 Background 1

1.2 Problem Definition 2

1.3 Scope and Motivation 2

1.4 Objectives 3

1.5 Challenges 3

1.6 Assumptions 4

1.7 Societal / Industrial Relevance 4

1.8 Organization of the Report 5

2 Literature Survey 7

2.1 Satellite Imagery for Deforestation Prediction using Deep Learning 7

2.2 Object Detection in Aerial Images: A Large-Scale Benchmark and Challenges 9

2.3 Machine Learning Algorithms for Satellite Image Classification Using Google Earth Engine and Landsat Satellite Data: Morocco Case Study 10

2.3.1 Minimum Distance Classification Method 11

2.4 Unsupervised change detection analysis in satellite image time series 12

2.5 Geographical Knowledge-Driven Representation Learning for Remote Sensing Images 15

2.6 Summary and Gaps Identified . . . 18
2.6.1 Summary . . . 18
2.6.2 Gaps Identified . . . 19

3 Requirements **21**
3.1 Hardware and Software Requirements . . . 21
3.2 Functional Requirements (Numbered List/ Description in Use Case Model) 22

4 System Architecture **23**
4.1 System Overview . . . 23
4.2 Architectural Design . . . 25
4.3 Module Division . . . 26
4.4 Work Schedule - Gantt Chart . . . 27

5 System Implementation **28**
5.1 Datasets Identified . . . 29
5.2 Proposed Methodology/Algorithms . . . 30
5.2.1 Random Forest Regressor . . . 31
5.3 User Interface Design . . . 33
5.4 Database Design . . . 33
5.5 Description of Implementation Strategies . . . 34

6 Results and Discussions **36**
6.1 Overview . . . 36
6.2 Testing . . . 37
6.3 Quantitative Results . . . 39
6.4 Graphical Analysis . . . 40
6.4.1 Comparison of Yearly Forest Loss Area in India, Zambia, Brazil, and Indonesia . . . 40
6.5 Discussion . . . 43

7 Conclusions & Future Scope **44**

References **45**

List of Publications **46**

Appendix A: Presentation **47**

Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes **59**

Appendix C: CO-PO-PSO Mapping **64**

List of Abbreviations

1. SVM: Support Vector Machine
2. RF: Random Forest
3. KNN: K-Nearest Neighbors
4. CNN: Convolutional Neural Network
5. GEE: Google Earth Engine
6. CART: Classification and Regression Trees
7. MD: Minimum Distance
8. GTB: Gradient Tree Boost
9. GeoKR: Geographical Knowledge-driven Representation
10. RMSE: Root Mean Square Error
11. SVR: Support Vector Regression
12. LSTM: Long short-term memory
13. XGBoost: Extreme Gradient Boosting

List of Figures

1.1 The Machine Learning Lifecycle . . . 2

2.1 Architecture . . . 8
2.2 The Machine Learning Lifecycle . . . 9
2.3 Number of instances per image among DOTA and general object detection datasets . . . 10
2.4 Workflow recommended in this paper for machine learning algorithms. . . . 12
2.5 Proposed framework . . . 14
2.6 Bitemporal change detection algorithm . . . 14
2.7 Details on proposed method (GeoKR). (a) Process of using geographical knowledge to provide supervision for representation learning and network pretraining. (b) Process of fine-tuning on the pretraining model for downstream tasks. . . . 16
2.8 Supervision Obtain Process With Geographical Knowledge . . . 17
2.9 Details about proportion of different categories in the Levir-KR dataset. . . 17

4.1 System Architecture . . . 25
4.2 Sequence Diagram . . . 25
4.3 Gantt Chart-Phase 1 . . . 27
4.4 Gantt Chart-Phase 1 . . . 27

6.1 The Map . . . 37
6.2 Prediction Page . . . 37
6.3 Home Page . . . 38
6.4 Trend Analyzer . . . 38
6.5 Login Page . . . 39
6.6 Yearly Forest Loss-Brazil . . . 41
6.7 Yearly Forest Loss-India . . . 41

6.8 Yearly Forest Loss-Indonesia 42
6.9 Yearly Forest Loss-Zambia 42

List of Tables

2.1 Comparitive advantages and disadvantages 18

6.1 Comparison between different models 39

Chapter 1

Introduction

Our project addresses the critical issue of deforestation through an innovative system that integrates a user-friendly interface, a powerful Deforestation Predictor, and an insightful Trend Analyzer. Leveraging the Random Forest Classifier algorithm, our solution provides precise predictions of future deforestation intensity and visually represents historical trends. This project contributes a valuable tool for environmentalists, policymakers, and researchers, fostering a proactive and informed approach to deforestation monitoring and management. Through this comprehensive analysis, we aim to make a meaningful impact on environmental conservation and sustainable resource management.

1.1 Background

Deforestation has emerged as a pressing global concern, with profound implications for climate change, biodiversity, and ecosystem stability. As human activities continue to encroach upon natural habitats, the need for effective deforestation monitoring and management becomes paramount. Current scenarios underscore a disturbing trend of escalating deforestation rates, threatening the delicate balance of our planet's ecosystems.

The importance of addressing deforestation lies in its far-reaching consequences, including climate change acceleration, loss of biodiversity, and disruptions to local communities dependent on forests. Traditional monitoring methods often fall short in providing real-time, comprehensive insights. This project seeks to bridge this gap by introducing an advanced system that combines machine learning algorithms, a user-friendly interface, and proactive prediction capabilities.

The significance of our project is twofold. Firstly, it addresses the urgency of developing innovative tools for accurate deforestation prediction beyond 2024, enabling timely interventions to curb unsustainable practices. Secondly, the Trend Analyzer component offers a retrospective view, helping stakeholders understand historical patterns for more informed decision-making.

In light of these considerations, our project aims to contribute to the global efforts towards sustainable forestry practices and environmental conservation. By providing a multifaceted approach to deforestation analysis, we strive to empower environmentalists, policymakers, and researchers with the tools necessary to tackle the complexities of deforestation in a rapidly changing world.

1.2 Problem Definition

The aim of this project is to develop an advanced deforestation monitoring system using machine learning algorithms, specifically a Random Forest Classifier, to predict future deforestation intensities beyond 2024. The system will provide users with accurate and proactive insights, aiding in the timely identification and mitigation of deforestation in vulnerable regions.

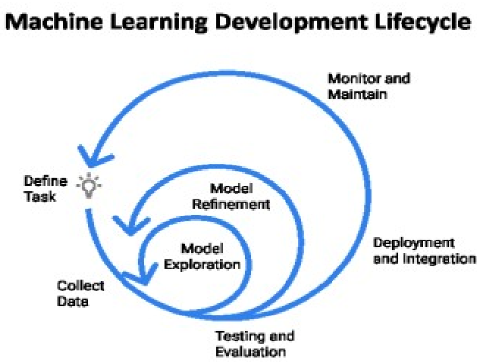


Figure 1.1: The Machine Learning Lifecycle

1.3 Scope and Motivation

The scope of this project encompasses the development of a comprehensive deforestation analysis system with a user-friendly interface, featuring a Deforestation Predictor and Trend Analyzer modules. The system's applicability extends to diverse user groups, including environmentalists, policymakers, and researchers. It focuses on addressing the critical need for accurate deforestation predictions beyond the year 2024, enabling timely intervention and sustainable resource management. Additionally, the Trend Analyzer component broadens the scope by providing historical insights, aiding stakeholders in understanding deforestation patterns over selected timeframes.

The motivation behind this project stems from the escalating threats posed by deforestation to global ecosystems. The pressing need for an advanced monitoring system arises from the limitations of existing methods in providing real-time and proactive insights. By leveraging machine learning algorithms, our system seeks to empower users with precise predictions and historical trend analyses, fostering a more informed and proactive approach to deforestation management. Ultimately, the motivation is rooted in contributing to global conservation efforts and providing a tool that can positively impact environmental sustainability.

1.4 Objectives

- Develop a user-friendly interface with login/signup functionality for easy access to the deforestation analysis system.
- Implement a Deforestation Predictor module utilizing the Random Forest Classifier algorithm to provide accurate and proactive predictions of deforestation intensities for selected years beyond 2024.
- Integrate a Trend Analyzer module that visually represents historical deforestation patterns over user-selected timeframes.
- Establish a secure and efficient data handling mechanism to ensure user privacy and maintain the confidentiality of sensitive information.

1.5 Challenges

Data Quality and Availability: The success of our machine learning model heavily relies on the quality and diversity of the dataset. Addressing missing values and ensuring thorough pre processing are crucial to enhance the accuracy and reliability of deforestation predictions.

Errors in Land Cover Classification: Misinterpretations stemming from errors in land cover classification pose a significant challenge. Ensuring precise classification is imperative to accurately identify and analyze deforestation events, reinforcing the robustness of our system's predictive and analytical capabilities.

1.6 Assumptions

- Low Cloud Cover in Satellite Images: We assume that the availability of satellite images with low cloud cover is consistent throughout the project. Limited cloud cover ensures clear visibility, allowing for accurate analysis of land cover changes and precise identification of deforestation events.

- Sufficient Spatial Resolution of Landsat: We assume that the spatial resolution provided by Landsat is adequate for effectively capturing and identifying changes in land cover. This resolution enables the detection of subtle variations on the Earth's surface, ensuring that deforestation events can be distinguished and analyzed with precision.

- Accurate Labeling of Land Cover Classes: It is assumed that the land cover classes in the satellite images are accurately labeled. Precise and reliable labeling is crucial for training the machine learning model, ensuring its ability to correctly identify and classify different land cover types, including areas undergoing deforestation.

1.7 Societal / Industrial Relevance

The project holds significant relevance for both societal and industrial applications. On a societal level, the deforestation analysis system provides a valuable tool for environmentalists, policymakers, and researchers engaged in sustainable resource management. It contributes to the global effort to monitor and combat deforestation, helping to preserve biodiversity, mitigate climate change, and ensure the well-being of communities dependent on forest ecosystems.

From an industrial perspective, the system's predictive capabilities offer benefits to industries involved in responsible resource utilization, such as the forestry and agriculture sectors. By proactively identifying potential deforestation hotspots, these industries can adopt more sustainable practices, aligning with environmental regulations and corporate social responsibility goals. Additionally, the technology-driven approach enhances efficiency in land management and planning, contributing to more informed decision-making processes.

In summary, the societal and industrial relevance of this project lies in its potential to foster environmental conservation, sustainable resource utilization, and responsible land management practices. By providing actionable insights, the deforestation analysis system addresses pressing global challenges and aligns with the broader goals of creating a more sustainable and ecologically conscious society.

1.8 Organization of the Report

The report unfolds with an introduction that lays the groundwork for our deforestation analysis system, delving into the background, problem statement, scope, objectives, challenges, assumptions, societal/industrial relevance, and the overall organization of the report. Following this, the literature survey navigates through relevant papers, condensing insights from five key studies while highlighting gaps in existing research.

The methodology chapter meticulously details the data collection, preprocessing, feature extraction, classification, and postprocessing steps, providing a comprehensive understanding of our approach. The subsequent chapter unveils the system architecture, offering an overview, architectural design, module division, and a Gantt chart delineating the work schedule. The systems and implemenations chapter discusses the technical aspects related to the development and deployment of our deforestation analysis and predictor. The results and discussions chapter presents the outcomes of our project and discuss their implications.

The concluding chapter synthesizes findings and outlines the future scope of the project. Finally, the report concludes with a comprehensive references section, acknowledging the scholarly contributions that informed our project's development. This organized structure ensures a coherent and thorough exploration of our deforestation analysis system from introduction to implementation and future prospects.

In conclusion, this introductory chapter has set the stage for our deforestation analysis project, emphasizing the critical need for innovative tools in addressing the escalating global issue of deforestation. By integrating a user-friendly interface, a powerful Deforestation Predictor, and a comprehensive Trend Analyzer, our system leverages the Random Forest Classifier algorithm to contribute precise predictions and historical trend analyses. The multifaceted approach aims to empower environmentalists, policymakers, and researchers in making informed decisions for effective deforestation monitoring and management.

As we delve into subsequent chapters, the literature survey will provide a condensed overview of relevant research, guiding our methodology. The detailed methodology will elucidate our approach, from data collection to postprocessing. The system architecture chapter will present an insightful view of the project's technical design and organizational structure. The implementations of our web application is discussed in the System Implementation chapter. The final results are shown in the Results and Discussions chapter. Finally, the concluding chapter will encapsulate key findings, outline future prospects, and reaffirm the societal and industrial significance of our deforestation analysis system. Through this comprehensive exploration, we aspire to contribute meaningfully to environmental conservation and sustainable resource management.

Chapter 2

Literature Survey

2.1 Satellite Imagery for Deforestation Prediction using Deep Learning

The methodology employed in this paper revolves around the application of deep learning techniques, particularly Deep Convolutional Neural Networks (CNNs), for predicting and detecting deforestation using satellite imagery. The introduction emphasizes the significance of deforestation in climate change and underscores the need for detection and prevention, prompting the use of advanced machine learning algorithms applied to satellite data.

In terms of machine learning methods, the paper briefly mentions traditional approaches such as Support Vector Machines and Naive Bayes but underscores a preference for deep learning techniques due to their reliability and ease of training. The chosen deep learning method is the Deep CNN, known for capturing robust representations and being relatively straightforward to train.

Data collection primarily relies on satellite imagery, with data sourced from various open-source platforms, including a Satellite Imagery Dataset from Kaggle, totaling around 40,000 images. Data preprocessing involves normalization to a range between 0 and 1 and resizing to ensure uniform dimensions for effective model training.

Model training encompasses the splitting of preprocessed data into training and test datasets. The paper discusses two training options: training from scratch and training using a pre-trained model. The latter approach is chosen, with mention of models like MobileNet, DenseNet, AlexNet, ResNet, VGG19, and EfficientNet. Fine-tuning is employed, freezing some layers and training others on target data.

Model testing involves validation on a separate test dataset, considering metrics like accuracy and loss. Analysis of variance and bias is conducted by comparing train and validation accuracies to address overfitting and underfitting concerns.

The deployment phase briefly touches upon various deployment environments such as mobile, web, robots, and devices like Raspberry Pi.

In the evaluation section, the model's performance is assessed using metrics like accuracy and loss, with figures representing the outcomes for both the original model and the ResNet pre-trained model provided.

The paper concludes by highlighting the detection capabilities of the developed model in identifying deforestation in satellite images. It underscores the advantages and limitations of the model and suggests potential directions for future research, including exploring models requiring less data, reducing model size, implementing segmentation models, applying data augmentation techniques, and experimenting with ensemble learning. The references include specific research papers, TensorFlow documentation, and Kaggle datasets.

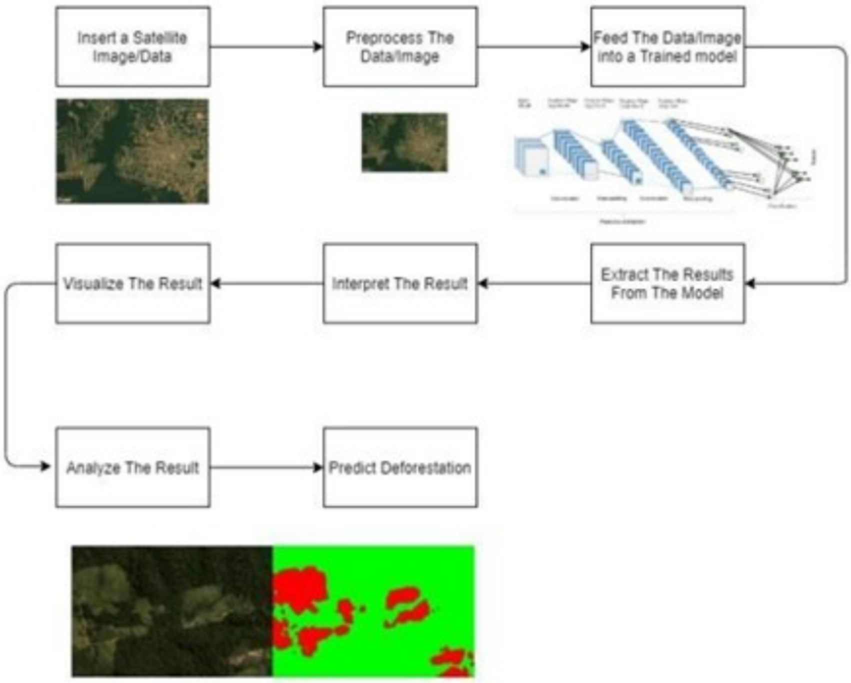


Figure 2.1: Architecture

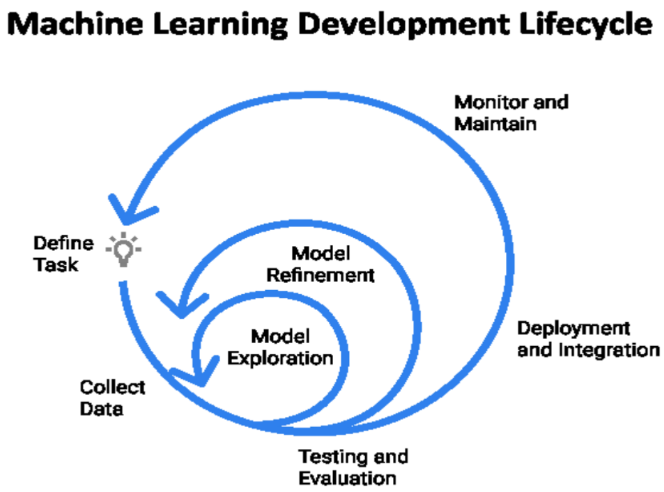


Figure 2.2: The Machine Learning Lifecycle

2.2 Object Detection in Aerial Images: A Large-Scale Benchmark and Challenges

The proposed methodology in this research aims to address the unique challenges posed by object detection in aerial images. Leveraging a large-scale benchmark dataset, the methodology employs a state-of-the-art convolutional neural network (CNN) architecture tailored for aerial imagery analysis. The dataset is meticulously curated to encompass diverse scenarios and object classes, ensuring a comprehensive evaluation of the detection model. The preprocessing stage involves data augmentation techniques to enhance the model's robustness and adaptability to real-world variations in lighting, perspective, and environmental conditions. To facilitate model training, a transfer learning approach is employed, initializing the CNN with weights pre-trained on a relevant large-scale dataset. The core detection model incorporates feature pyramid networks (FPN) to effectively capture multi-scale information, essential for identifying objects of varying sizes within aerial imagery. A region-based convolutional neural network (R-CNN) framework is employed for its ability to accurately localize and classify objects. To mitigate class imbalance challenges inherent in aerial imagery, a carefully designed loss function is introduced, giving higher weight to minority classes. The model is fine-tuned using a combination of region proposal network (RPN) and non-maximum suppression (NMS) to refine detection results and reduce false positives. The comprehensive evaluation of the proposed methodology involves benchmarking against existing state-of-the-art approaches, considering metrics such as precision, recall, and F1 score. The results demonstrate the efficacy of the proposed method in achieving accurate and robust object detection in aerial images across diverse scenarios.

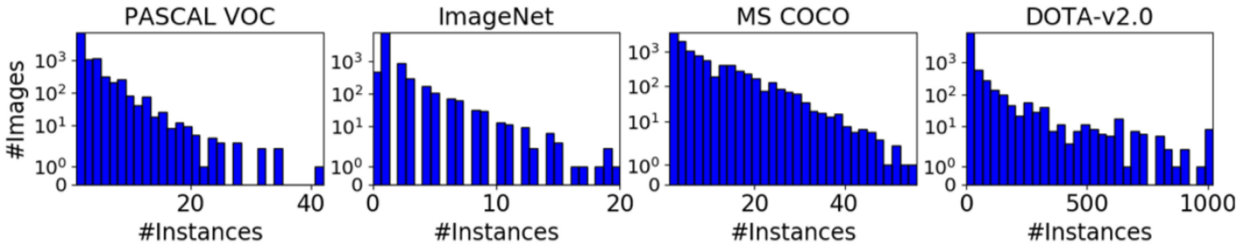


Figure 2.3: Number of instances per image among DOTA and general object detection datasets

2.3 Machine Learning Algorithms for Satellite Image Classification Using Google Earth Engine and Landsat Satellite Data: Morocco Case Study

1. Support Vector Machine (SVM): Support Vector Machine (SVM) is a supervised machine learning algorithm used for classification and regression tasks. It works by finding the optimal hyperplane that best separates the data into different classes. SVM is effective in handling high-dimensional data and is particularly useful when dealing with complex datasets with a clear margin of separation between classes. In the context of land cover classification, SVM can be applied to accurately delineate different land cover types based on spectral and spatial features extracted from satellite imagery.
2. Random Forest (RF): Random Forest is an ensemble learning method that operates by constructing multiple decision trees during training and outputting the class that is the mode of the classes of the individual trees. It is known for its robustness and ability to handle large datasets with high dimensionality. In the context of land cover classification, Random Forest can effectively capture the complex relationships between spectral bands and land cover classes, leading to accurate classification results.
3. Classification and Regression Trees (CART): Classification and Regression Trees (CART) is a decision tree learning algorithm that is used for both classification and regression tasks. It recursively partitions the data into subsets based on the value of the input features. CART is known for its interpretability and ability to handle both

numerical and categorical data. In the context of land cover classification, CART can be used to create a set of decision rules based on spectral and spatial features, enabling the classification of different land cover types.

4. Minimum Distance (MD): The Minimum Distance (MD) method is a simple and intuitive classification algorithm that assigns each pixel to the class with the closest spectral signature. It calculates the Euclidean distance between the spectral values of each pixel and the class centroids, and assigns the pixel to the class with the minimum distance. In the context of land cover classification, the MD method can be used to quickly classify pixels based on their spectral similarity to predefined class centroids.

5. Decision Tree (DT): Decision Tree is a non-parametric supervised learning method used for classification and regression tasks. It recursively splits the data into subsets based on the value of the input features, creating a tree-like model of decisions. Decision Tree is known for its interpretability and ability to handle both numerical and categorical data. In the context of land cover classification, Decision Tree can be used to create a hierarchical set of decision rules based on spectral and spatial features, facilitating the classification of different land cover types.

6. Gradient Tree Boost (GTB): Gradient Tree Boosting is an ensemble learning method that builds a series of decision trees, where each tree corrects the errors of the previous one. It is known for its ability to handle complex relationships in the data and its robustness against overfitting. In the context of land cover classification, Gradient Tree Boosting can effectively capture the complex relationships between spectral bands and land cover classes, leading to accurate classification results. These methods, when applied in the context of land cover classification using satellite imagery and machine learning algorithms, contribute to the accurate delineation and mapping of different land cover types, providing valuable information for urban planning and environmental monitoring.

2.3.1 Minimum Distance Classification Method

The most significant method used in the paper is the Minimum Distance (MD) classification algorithm. This method is particularly noteworthy due to its simplicity and efficiency in assigning pixels to specific land cover classes based on their spectral similarity to predefined class centroids. The MD method calculates the Euclidean distance between the spectral values of each pixel and the class centroids, and assigns the pixel to the class with the minimum distance. In the context of land cover classification, the MD method offers a straightforward yet effective approach to quickly and accurately classify pixels, making it a valuable tool for mapping and monitoring land cover in urban areas. The study's findings, which highlight the MD algorithm's high accuracy in classifying land cover in Morocco, underscore its significance as a reliable and efficient method for satellite image classification and urban planning applications.

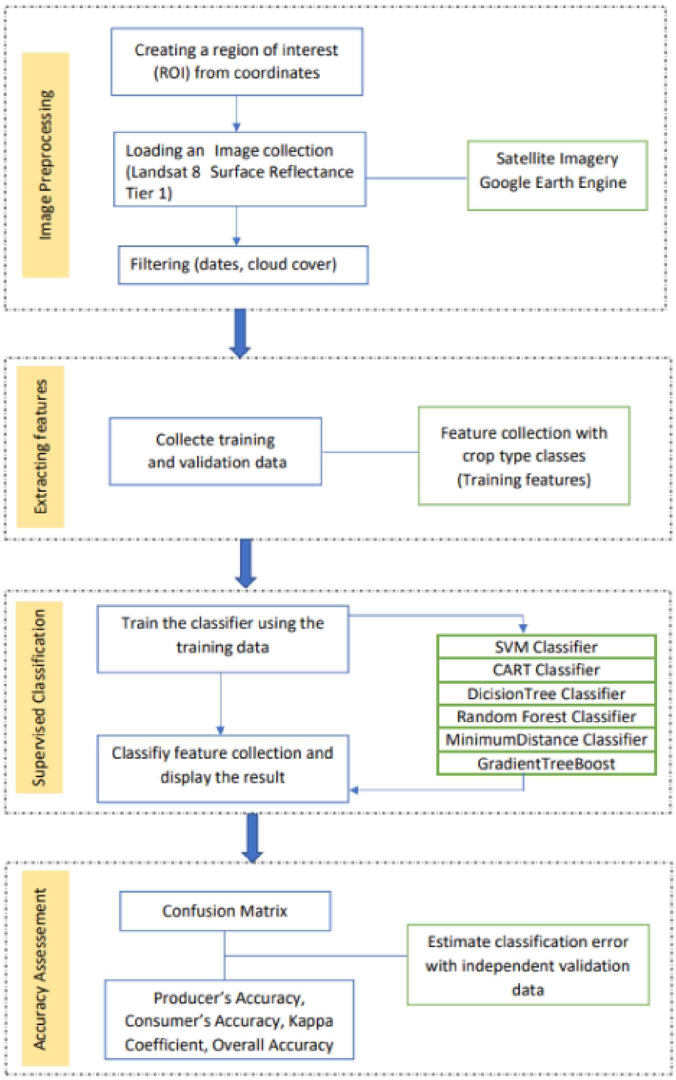


Figure 2.4: Workflow recommended in this paper for machine learning algorithms.

2.4 Unsupervised change detection analysis in satellite image time series

1. Bitemporal Change Detection: - The bitemporal change detection algorithm is designed to identify nontrivial changes in the satellite image time series. The algorithm

compares consecutive images in the time series and generates binary change maps for each pair of images. The change maps highlight areas where significant changes have occurred between the two images. The algorithm is tailored to handle less numerous and unique changes that are often considered as outliers by traditional clustering algorithms. The approach is designed to be adaptable to various datasets with different temporal characteristics, and it does not depend on the temporal resolution of the satellite image

2. Image Segmentation: - The extracted change areas from the bitemporal change detection step are then segmented using a tree-merging bottom-up segmentation algorithm. This algorithm merges adjacent pixels with similar spectral properties into larger segments. The segmentation parameters are chosen empirically to produce relevant segments for both single images and concatenated images. Additionally, reference adjacent objects with similar spectral properties are manually chosen for segmentation, ensuring that the change objects extracted from concatenated images are relevant to single-image objects.

3. Radiometric Normalization: - To ensure homogeneous and comparable spectral values across the dataset, both satellite image time series are radiometrically normalized using an algorithm based on histogram analysis of pixel distributions. This normalization step aims to standardize the spectral values, enabling consistent analysis and comparison of the images. The normalization algorithm is designed to be robust to outliers and to preserve the relative spectral differences between the images .

4. Graph-Based Clustering: - The framework incorporates graph-based techniques for clustering the detected change processes. The multitemporal context of the change processes is analyzed to construct different change processes, which are further grouped into clusters. The graphs constructed with this method provide a global description of the detected change processes, allowing for the interpretation of various subprocesses. The approach is designed to be fully unsupervised and has shown promising results on real-life datasets. However, it is noted that for unbalanced datasets, smaller classes may not be well separated from the majority ones. The clustering algorithm is based on a neural network autoencoder that extracts features from the change areas and a graph-based clustering algorithm that groups the extracted features into clusters .

5. Application and Contributions: - The proposed method has a wide range of potential applications, including urban evolution studies, land cover analysis, and ecological assessments. The contributions of the work are summarized, emphasizing the novelty of

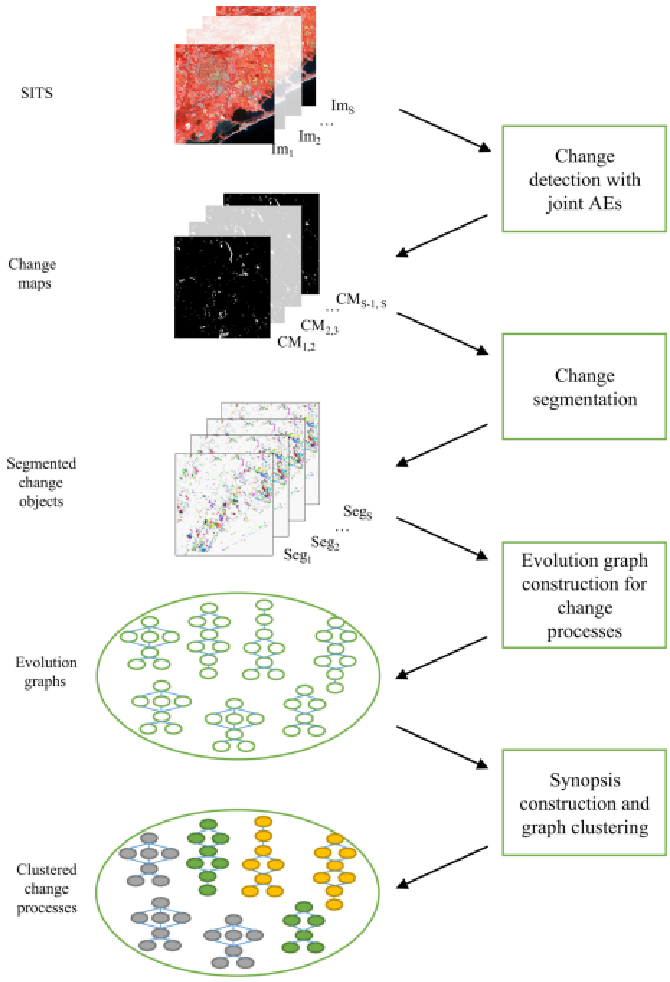


Figure 2.5: Proposed framework

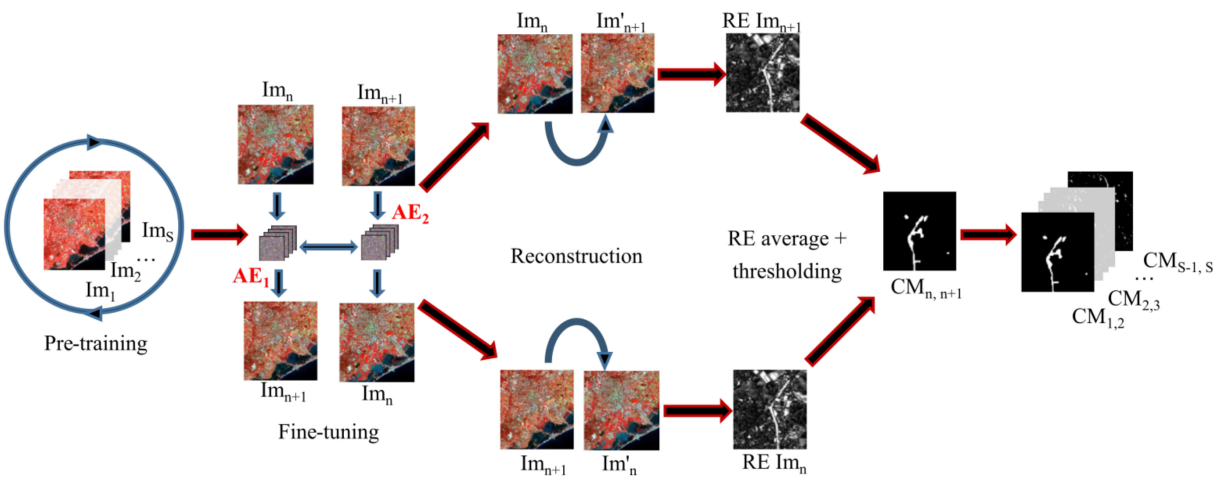


Figure 2.6: Bitemporal change detection algorithm

the unsupervised framework that combines change detection and clustering of subsequent change graphs. The approach is designed to be adaptable to various datasets with different temporal characteristics and does not depend on the temporal resolution of the satellite image time series. The authors note that the proposed method can be applied to short time series and has the potential to provide valuable insights into spatiotemporal phenomena .

2.5 Geographical Knowledge-Driven Representation Learning for Remote Sensing Images

The paper leverages geographical knowledge, such as global land cover products and geographical location, to propose a novel method for representation learning and network pretraining in remote sensing images: Geographical Knowledge-driven Representation (GeoKR) learning.

• In order to enhance network performance and lessen the dependency on annotated data, the suggested approach incorporates geographic information to help tackle the problem of efficiently employing the enormous quantity of unlabeled remote sensing images.

• Using mean-teacher networks as the foundation, the study creates an effective pretraining framework that synchronizes the network's image representations with the knowledge representations produced by geographic knowledge.

• To effectively support network pretraining, the authors build a pretraining dataset called Levir-KR, which consists of 1,431,950 remote sensing images from Gaofen satellites with varying resolutions.

• The effectiveness of the suggested GeoKR method is assessed through experimental validation on remote sensing datasets, showing how it enhances the performance of downstream tasks like object detection, semantic segmentation, scene classification, and cloud/snow detection.

• The experimental results demonstrate that the suggested approach can successfully lessen the workload associated with data annotation and outperforms current pretraining techniques like ImageNet pretraining and self-supervised representation learning.

• The study concludes that by offering a novel solution that tackles the difficulties associated with the availability of annotated data and improves the use of remote sensing images in a variety of applications, the suggested GeoKR method can further the development of remote sensing image processing.

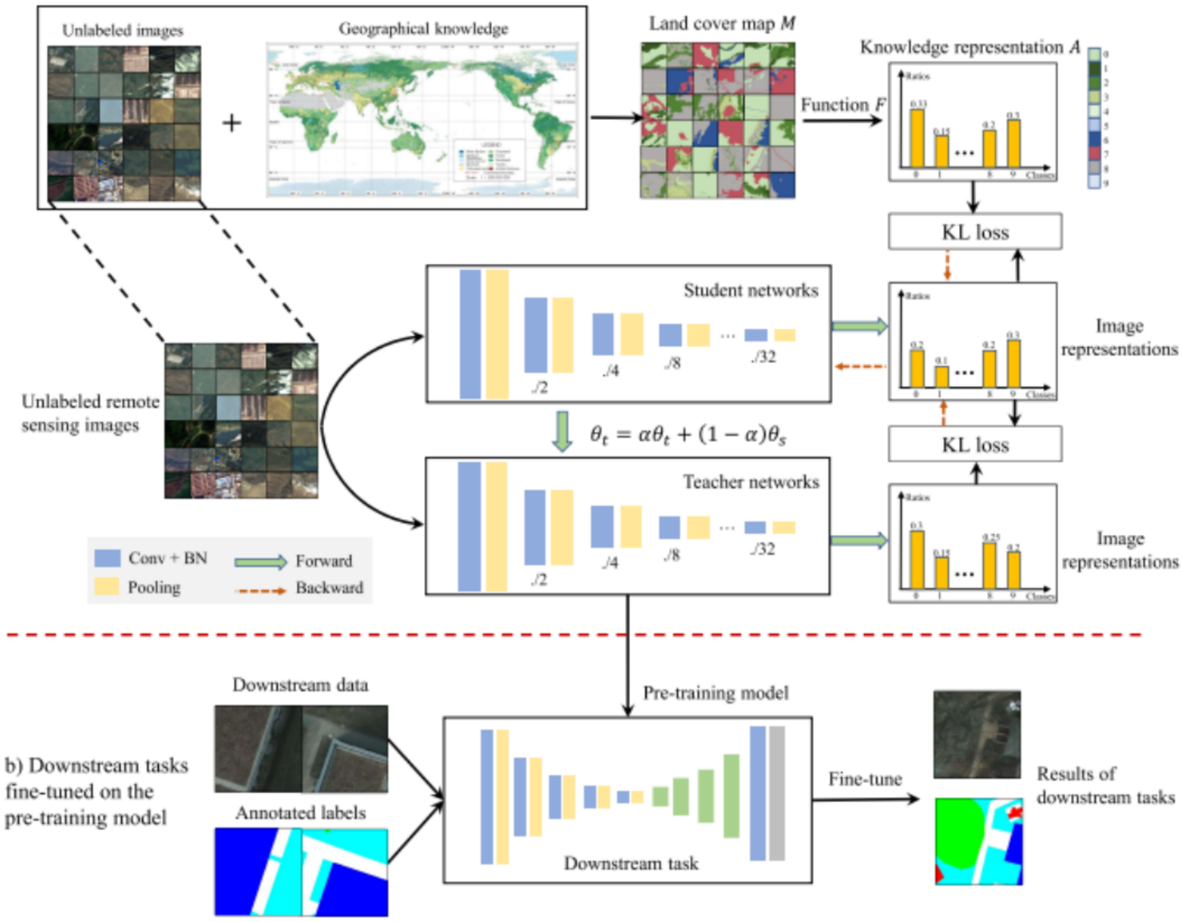


Figure 2.7: Details on proposed method (GeoKR). (a) Process of using geographical knowledge to provide supervision for representation learning and network pretraining. (b) Process of fine-tuning on the pretraining model for downstream tasks.

Algorithm 1 Supervision Obtain Process With Geographical Knowledge

1: **Input:** remote sensing image I of size s and its geographical location GT_I, GlobeLand30 product G and geographical location GT_g^j of jth area g_j, the number of area N.
2: **Output:** knowledge representation vector A of image I.
    ▷ firstly, determine to which area the image I belongs.
3: **for all** $j \in \{1, 2, \ldots, N\}$ **do**
4:     Determine whether GT_I is in GT_g^j.
5:     **if** $\mathrm{GT}_I \in \mathrm{GT}_g^j$ **then**
6:         $\mathrm{GT}_g^m \leftarrow \mathrm{GT}_g^j$
7:     **end if**
8: **end for**
    ▷ secondly, calculate the relative coordinate of the image in the area.
9: calculate $x_{\mathrm{left}}, x_{\mathrm{top}}, x_{\mathrm{right}}, x_{\mathrm{bottom}}$ using Eq. 1 - Eq. 4.
    ▷ finally, the proportion of different land covers is counted as the knowledge representation A
10: land cover map $M \leftarrow g_j[x_{\mathrm{top}} : x_{\mathrm{bottom}}, x_{\mathrm{left}} : x_{\mathrm{right}}]$
11: calculate knowledge representation vector A
12: **return** A

Figure 2.8: Supervision Obtain Process With Geographical Knowledge

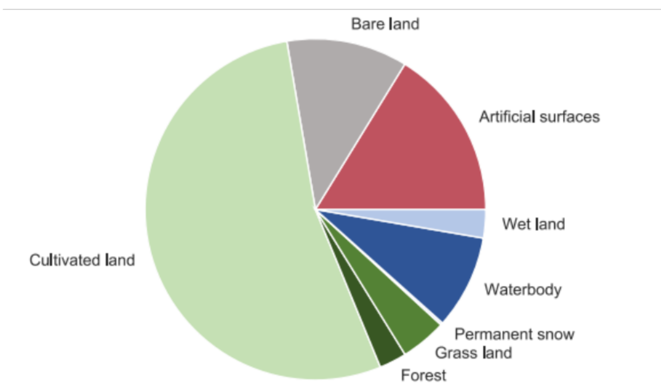


Figure 2.9: Details about proportion of different categories in the Levir-KR dataset.

2.6 Summary and Gaps Identified

2.6.1 Summary

Table 2.1: Comparitive advantages and disadvantages

Paper	Method	Advantages	Disadvantages
[1]	Deep convolutional neural networks	High-dimensional, Accurate	Computationally Intensive
[2]	Dataset Expansion,Code Library creation and Algorithm Evaluation	- Large-scale dataset - Code library and website	-Limited generalizability - Limited scope
[3]	GEE platform and machine learning algorithms	-Use of advanced machine learning algorithms -Accurate land cover information	-Limited ground truth data -Limited spatial resolution
[4]	End-to-End approach for change detection and clustering for satellite image time series ,which combines deep learning with graph based approaches	-Reference data for model training is not required - Method is able to identify nontrivial changes	-Difficult to implement as it requires complex deep-learning algorithms
[5]	GeoKR, utilizes geographical knowledge to extract supervision information and construct an efficient representation learning framework for removing the influence of noise labels.	-Geographical knowledge to improve network performance. - Reducing the demand for annotated data and outperforming existing pretraining methods.	-Limited generalizability -Difficult to implement

2.6.2 Gaps Identified

- Limited Exploration of Traditional Methods: - The paper briefly mentions traditional methods such as Support Vector Machines and Naive Bayes but does not delve into a detailed comparison or analysis of their performance against the deep learning approach. A more in-depth exploration of the strengths and weaknesses of both traditional and deep learning methods could provide a more nuanced understanding.

- Lack of Detailed Model Architecture Description: - The paper mentions the use of various pre-trained models, including MobileNet, DenseNet, AlexNet, ResNet, VGG19, and EfficientNet, but lacks a detailed description of the specific architecture used. Providing more details about the selected architecture, layer configurations, and reasoning behind the choice would enhance the clarity of the methodology.

- Insufficient Discussion on Hyperparameter Tuning: - While the paper touches upon factors like learning rate, optimizer, loss function, epochs, and batch size, it does not provide a detailed discussion on the process of hyperparameter tuning. A more thorough exploration of the hyperparameter tuning process and its impact on model performance would strengthen the methodology.

- Limited Discussion on Model Generalization: - The paper lacks a comprehensive discussion on how well the developed model generalizes to unseen data or different geographical locations. Addressing the model's ability to generalize beyond the training data is crucial for assessing its real-world applicability.

- Absence of Results on Different Satellite Image Datasets: - The paper mentions the use of a Satellite Imagery Dataset from Kaggle, but it does not present results or insights based on experiments with other satellite image datasets. Evaluating the model's performance across diverse datasets would provide a more robust assessment of its capabilities.

- Inadequate Exploration of Ensemble Learning: - While ensemble learning is briefly mentioned as a potential direction for future work, the paper does not explore or experiment with ensemble methods in the current research. A more in-depth analysis of ensemble learning techniques and their impact on prediction accuracy could be beneficial.

- Limited Discussion on Model Interpretability: - The paper lacks a discussion on the interpretability of the developed model. Understanding how the model makes decisions and identifying which features contribute to deforestation detection could enhance the transparency and trustworthiness of the model.

In conclusion,the first paper focuses on deforestation prediction using deep learning and satellite imagery. The authors employ advanced machine learning algorithms, including convolutional neural networks (CNNs) like ResNet and MobileNet, to identify deforested lands. The proposed architecture involves data gathering, preprocessing, training, testing, and deployment stages. The goal is to inform government agencies when deforestation surpasses a certain threshold. The second paper discusses machine learning algorithms for satellite image classification in Morocco. Algorithms such as Support Vector Machine (SVM), Random Forest (RF), and others are explored for accurate land cover delineation. The Minimum Distance (MD) classification method stands out for its simplicity and effectiveness in quickly classifying pixels based on spectral similarity. The third paper introduces an unsupervised change detection framework for satellite image time series, encompassing bitemporal change detection, segmentation, radiometric normalization, and graph-based clustering. This methodology aims to identify and analyze significant changes, providing adaptability to various datasets for urban planning and ecological assessments. The fourth paper introduces Geographical Knowledge-driven Representation (GeoKR) learning for remote sensing images. This approach incorporates geographic information to enhance network pretraining, reducing the reliance on annotated data. The GeoKR method is demonstrated to improve downstream tasks such as object detection and semantic segmentation, offering a novel solution for leveraging geographical knowledge in remote sensing image processing.

Chapter 3

Requirements

3.1 Hardware and Software Requirements

1. Hardware Requirements: - Processor: A multi-core processor with a minimum of 2.5 GHz clock speed or higher. - RAM: A minimum of 8 GB RAM for optimal performance during data processing and machine learning tasks. - Storage: Adequate storage space, preferably SSD, to accommodate large datasets and model training files. - Graphics Card: A dedicated GPU (Graphics Processing Unit) with CUDA support for efficient parallel processing, recommended for enhancing machine learning model training speed.

2. Software Requirements: - Operating System: Compatibility with Windows 10, Linux (Ubuntu, CentOS), or macOS for flexibility and ease of development. - Python: Latest version of Python 3.x for implementing machine learning algorithms and system functionalities. - Integrated Development Environment (IDE): Use of popular IDEs such as Jupyter Notebook, PyCharm, or VSCode for code development and experimentation. Google Earth Engine, Google Colab and Python Flask are also used.

3. External Data Sources: - Landsat Satellite Data: Access to Landsat satellite images with low cloud cover, ensuring clear visibility for accurate land cover analysis. - Deforestation Datasets: Relevant datasets containing labeled deforestation events for training and testing machine learning models.

Ensuring compatibility and adherence to these hardware and software requirements will optimize the performance and reliability of the deforestation analysis system. Regular updates and maintenance of software components are recommended to incorporate the latest features and security patches.

3.2 Functional Requirements (Numbered List/ Description in Use Case Model)

Functional Requirements:

1. User Registration and Authentication: Users can register for an account by providing a username, email, and password. Upon registration, users can log in using their credentials to access the application's features.

2. Deforestation Predictor: Users can select a point on a map of Para using latitude and longitude coordinates. They can then choose a future year (after 2024) to see the predicted amount of deforestation in square kilometers for that area.

3. Trend Analyzer: Users can select a year gap and a country (India, Zambia, Indonesia, or Brazil) to visualize deforestation patterns as a bar graph. This feature helps users understand historical deforestation trends.

4. User Profile Management: Users can view and edit their profile information, including username, email, and password. They can also delete their account if needed.

5. Data Visualization: The application provides interactive and visually appealing data visualizations, such as maps and graphs, to help users understand deforestation trends easily.

6.Admin Panel: An admin panel is available for administrators to manage user accounts, monitor system usage, and perform other administrative tasks. 7.Security: User passwords are stored securely using encryption techniques. The application follows best practices for data security to protect user information.

8.Responsive Design: The application is designed to be responsive, ensuring a seamless user experience across different devices and screen sizes.

9.Error Handling: The application provides informative error messages and handles errors gracefully to enhance user experience.

10.Performance Optimization: The application is optimized for performance, ensuring fast loading times and smooth user interactions.

These functional requirements collectively define the core features and capabilities of the deforestation analysis system, providing a user-centric and secure environment for effective monitoring and management.

Chapter 4

System Architecture

This Chapter deals with the Architecture Design of the model in detail.It also deals with different modules of the project and a brief description of the same.The chapter is concluded with a Gantt Chart,outlining the work schedule.

4.1 System Overview

- **Data Collection and Preprocessing:**
- Satellite Imagery Acquisition: Obtain high-resolution satellite imagery using platforms like Landsat, Sentinel, or other relevant sources.
- Training Data Generation: Collect labeled training data by manually annotating areas in the images as deforested or non-deforested.
- Image Preprocessing: Perform preprocessing steps, including cropping, resizing, and normalization of pixel values. Extract relevant features such as NDVI (Normalized Difference Vegetation Index) from the satellite imagery.
- **Training Dataset Splitting:**

Divide the labeled dataset into training and validation sets. This ensures that the model's performance can be evaluated on data it has not seen during training.

- **Random Forest Training:**
- Feature Selection: Choose relevant spectral bands and derived indices as features for training the Random Forest model.
- Random Forest Configuration: Set hyperparameters for the Random Forest classifier, such as the number of trees, maximum depth, and other settings.

- Model Training: Use the training dataset to train the Random Forest classifier. During training, the algorithm builds multiple decision trees based on random subsets of features and data.

- **Model Evaluation:**

- Validation: Assess the model's performance on the validation set using metrics like accuracy, precision, recall, and F1 score.

- Hyperparameter Tuning: Fine-tune the Random Forest hyperparameters to optimize the model's performance.

- **Prediction on Satellite Images:**

- Image Classification: Apply the trained Random Forest model to classify each pixel in the satellite images as deforested or non-deforested.

- Deforestation Map Generation: Create a deforestation map based on the classification results, visualizing areas with detected deforestation.

- **Post-Processing:**

- Noise Reduction: Implement post-processing techniques to remove small misclassifications or noise in the classification results.

- Image Enhancement: Apply filters or algorithms to enhance the clarity of the deforestation map.

- **Result Analysis:**

- Accuracy Assessment: Evaluate the accuracy and reliability of the deforestation detection results.

- Ground Truth Validation: If available, compare the results with ground truth data to validate the accuracy of the model.

- **Monitoring and Updating:**

- Periodic Monitoring: Implement a monitoring system to regularly assess deforestation changes over time.
- Model Updating: Periodically update the Random Forest model with new labeled data to adapt to changing conditions and improve accuracy.

4.2 Architectural Design

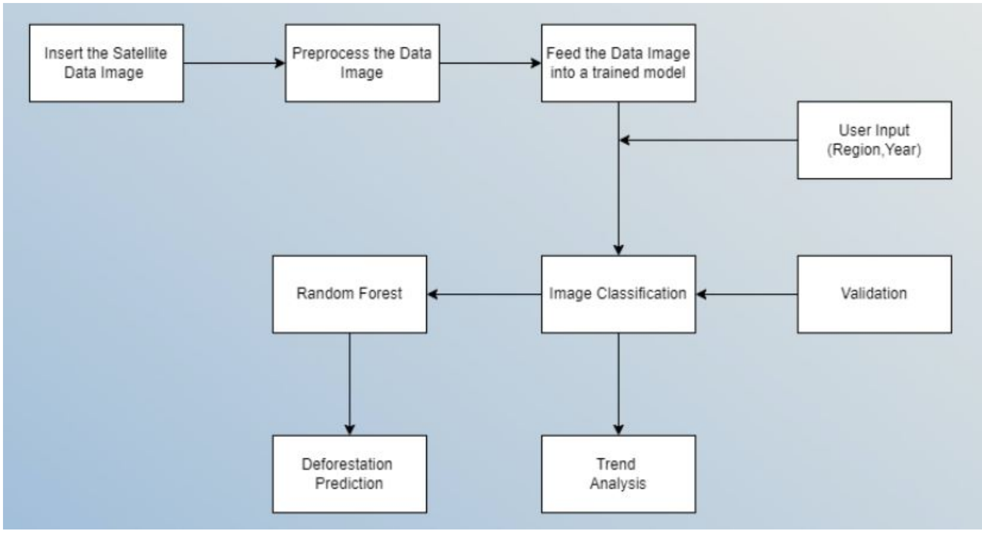


Figure 4.1: System Architecture

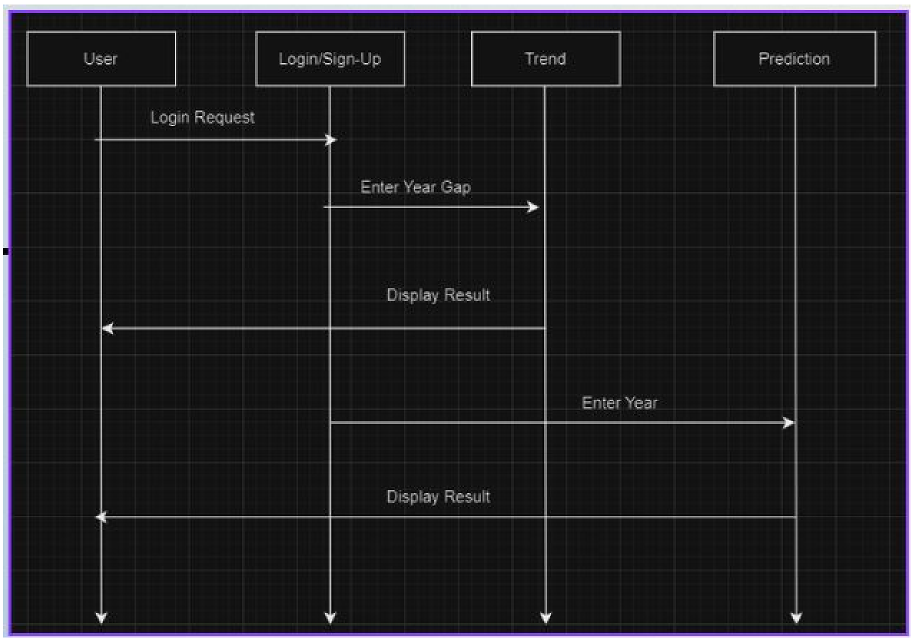


Figure 4.2: Sequence Diagram

4.3 Module Division

- **Map, Dataset Collection and Model Training and frontend**- Shaun Tojan
  - The frontend development of the project, which involves creating the user interface and ensuring a smooth user experience.
  - Involved in the map component of the project, working on integrating maps into the web application and allowing users to interact with geographical data.
- **Dataset Collection, Flask, Model Training and frontend** - Tijin T Babu
  - Setting up the web framework of the application.
- **Model Training, Flask, Random Forest Classifier and backend** - Richard Sherlin
  - Implementing the Random Forest machine learning algorithm for the project, which is used for predicting deforestation trends.
  - Setting up the web framework and backend functionality of the application.
- **Dataset Collection, Map, Model Training and Frontend** - Roy Rajesh
  - Involved in the map component of the project, working on integrating maps into the web application and allowing users to interact with geographical data.
  - Frontend development, Model training and Dataset collection.

4.4 Work Schedule - Gantt Chart

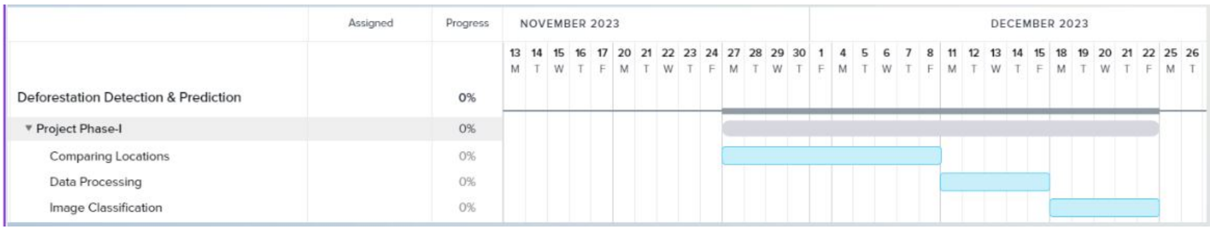


Figure 4.3: Gantt Chart-Phase 1

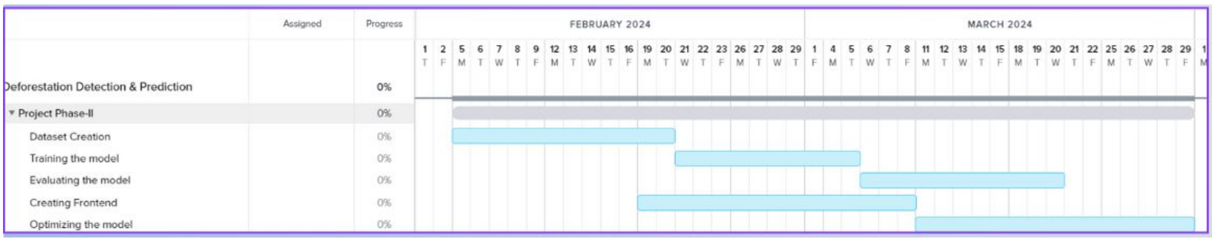


Figure 4.4: Gantt Chart-Phase 1

In summary, this chapter provided an in-depth exploration of the project's system architecture, including detailed diagrams illustrating the overall structure. The various modules crucial to the project were introduced, each serving specific functions within the system. These modules were further outlined, and responsibilities were assigned to team members for efficient development. Additionally, a Gantt chart was presented to depict the project's timeline, highlighting the planned schedule for module development, testing, and integration phases. This structured approach aims to ensure a systematic and well-managed progression through the project's implementation stages.

Chapter 5

System Implementation

In this chapter, we embark on a comprehensive journey towards designing an integrated system tailored for the deployment and utilization of machine learning algorithms. Our endeavor encompasses the identification of pertinent datasets, delineation of proposed methodologies and algorithms, meticulous user interface design, establishment of a robust database architecture, and elucidation of implementation strategies. Each facet contributes to the seamless functionality and efficacy of the overarching machine learning system.

Our journey begins with the critical task of dataset identification. Through a rigorous selection process, we identify datasets that encapsulate the essence of the problem domain, ensuring relevance, and reliability. These datasets serve as the cornerstone for subsequent model training and evaluation, empowering the machine learning system with the necessary input for informed decision-making and pattern recognition.

Subsequently, we delve into proposed methodologies and algorithms poised to unlock the latent potential within the identified datasets. Drawing upon a synthesis of established techniques and innovative approaches, our methodology aims to develop robust machine learning models capable of accurate prediction, classification, or clustering, depending on the application at hand.

Furthermore, we dedicate attention to user interface design, recognizing its pivotal role in facilitating user interaction and comprehension. Through intuitive layout, seamless navigation, and informative visualizations, our user interface endeavors to empower users to interact with machine learning models effectively, interpret results, and make informed decisions based on generated insights.

In parallel, we explore the intricacies of database design, laying the foundation for robust data management and model integration. Through meticulous schema design, indexing strategies, and optimization techniques, our database architecture aims to uphold data integrity, scalability, and performance, facilitating seamless interaction between the machine learning system and underlying data infrastructure.

Finally, we elucidate implementation strategies poised to transform conceptual designs into tangible realities. From model training and evaluation to deployment methodologies, each facet is meticulously orchestrated to foster a smooth transition from prototype to production, culminating in a functional system poised to harness the transformative power of machine learning across various domains and applications.

Through the convergence of dataset identification, methodology formulation, user interface design, database architecture, and implementation strategies, this chapter serves as a roadmap towards the realization of an integrated system poised to unlock the full potential of machine learning in addressing real-world challenges and driving innovation.

5.1 Datasets Identified

The dataset represents a compilation of geospatial data points exclusively sourced from the state of Pará, Brazil. Each entry in the dataset corresponds to a distinct geographic area within Pará, characterized by its longitude and latitude coordinates, alongside the year of data collection. This collection of approximately 4000 rows was meticulously crafted through manual classification in Google Earth Engine, highlighting the fusion of human expertise and technological capabilities in geospatial analysis.

Here's a concise overview of the key attributes of the dataset:

1. Area: Each data point pertains to a unique geographical region within the state of Pará, Brazil. These areas could encompass diverse landscapes, including forests, rivers, urban centers, agricultural regions, and protected areas, among others.

2. Longitude and Latitude: The dataset includes longitude and latitude coordinates for each area, pinpointing its precise location within Pará. Longitude indicates the east-west position relative to the Prime Meridian, while latitude denotes the north-south position relative to the Equator. These coordinates provide essential spatial references for geospatial analysis and mapping.

3. Year: The year column denotes the temporal dimension of the dataset, signifying the specific year when data was collected or analyzed for each area within Pará. This

temporal aspect enables the tracking of changes over time, facilitating temporal analysis and trend identification.

4. Manual Classification in Google Earth Engine: The dataset was meticulously curated through manual classification within the Google Earth Engine platform, focusing exclusively on areas within the state of Pará, Brazil. This process involved visual interpretation and classification of satellite imagery or other geospatial datasets to delineate distinct features or land cover classes within Pará's territory.

5.2 Proposed Methodology/Algorithms

To acquire a dataset from Google Earth Engine utilizing classification with a Random Forest classifier, first define the area of interest (AOI) and select relevant satellite imagery or geospatial datasets within the Google Earth Engine platform. Next, preprocess the data by cleaning, filtering, and preparing it for classification. Utilize the Random Forest classifier algorithm to perform supervised classification, where training samples are manually labeled to train the classifier to recognize different land cover classes within the AOI. Apply the trained classifier to the entire dataset to generate classified maps or raster images representing land cover types. Finally, extract the classified dataset from Google Earth Engine for further analysis, visualization, or integration into machine learning workflows. This process enables the creation of labeled datasets for training machine learning models and conducting land cover analysis or monitoring within the specified geographic region.

Dataset preprocessing is a crucial preparatory phase in the machine learning pipeline, involving a series of steps to clean, transform, and enhance raw data for optimal model training. This includes addressing missing values through imputation or deletion, scaling features to ensure uniformity across different scales, encoding categorical variables into numerical format, performing feature engineering to create informative features, detecting and treating outliers to prevent model distortion, and splitting the dataset into training, validation, and test sets for effective model evaluation. By meticulously preprocessing the dataset, we ensure that the machine learning algorithms receive high-quality input data, leading to improved model performance, accuracy, and generalization capabilities when making predictions or generating insights.

Different models that were trained :

1. XGBoost (Extreme Gradient Boosting): XGBoost is a powerful ensemble learning algorithm that sequentially builds decision trees to correct errors and improve predictions. It's highly efficient and effective for various tasks.

2. Random Forest: Random Forest constructs multiple decision trees using random subsets of the data and features. It aggregates predictions to enhance accuracy and is robust against overfitting.

3. Linear Regression: Linear Regression models the relationship between variables by fitting a linear equation to the data. It's simple, interpretable, and suitable for linearly related data.

4. SVR (Support Vector Regression): SVR is a variant of Support Vector Machines adapted for regression tasks. It finds the optimal hyperplane to minimize errors within a specified margin, effective for high-dimensional data.

5. LSTM (Long Short-Term Memory): LSTM is a type of recurrent neural network specialized in capturing long-term dependencies in sequential data. It's widely used in time series forecasting and natural language processing.

After training these different machine learning algorithms and calculating their RMSE (Root Mean Squared Error) values, the algorithm with the least RMSE was selected for prediction purposes. In this case, Random Forest was chosen, likely due to its robustness, ease of use, and strong performance across a variety of datasets. Its ability to handle both numerical and categorical features, as well as its resistance to overfitting, made it a suitable choice for your prediction task.

5.2.1 Random Forest Regressor

Here's a detailed explanation of Random Forest Regressor:

1. Ensemble Learning: Random Forest is an ensemble learning method, which means it combines the predictions of multiple individual models to produce a more accurate and robust final prediction. In the case of Random Forest, the individual models are decision trees.

2. Decision Trees: Decision trees are a type of supervised learning algorithm used for both classification and regression tasks. They break down the data into smaller subsets based on feature values, leading to a tree-like structure where each internal node represents a feature, each branch represents a decision based on that feature, and each leaf node represents the predicted outcome. Decision trees are prone to overfitting, especially when they become too deep or complex. Random Forest mitigates this issue by training multiple decision trees on random subsets of the data and features.

3. Random Forest Architecture: Random Forest consists of a collection of decision trees. During training, each tree is trained independently on a random subset of the training data (known as bagging or bootstrap aggregation) and a random subset of the features. The randomness introduced in both the data sampling and feature selection helps to decorrelate the individual trees, reducing the risk of overfitting and improving the model's generalization performance.

4. Prediction Process: To make a prediction with a Random Forest Regressor, each decision tree in the ensemble independently produces a prediction based on the input features. The final prediction is then calculated by aggregating the predictions of all the individual trees. For regression tasks, this typically involves averaging the predictions of all trees.

5. Advantages: Random Forest Regressor offers several advantages, including: Robustness to overfitting: By averaging the predictions of multiple trees, Random Forest reduces the risk of overfitting compared to individual decision trees. Handle large datasets: Random Forest can efficiently handle large datasets with many features and observations. Feature importance: It provides a measure of feature importance, allowing users to identify the most influential features in the prediction process.

6. Parameters: Random Forest Regressor has several hyperparameters that can be tuned to optimize performance, such as the number of trees in the forest, the maximum depth of each tree, and the minimum number of samples required to split a node.

5.3 User Interface Design

The user interface design of our project includes:

1. Home Page: Header with project name/logo and login/signup buttons. Main content area showing a brief overview of the project and its features. Footer with contact information and links to social media.

2. Login/Signup Page: Input fields for username/email and password. Separate tabs for login and signup options. "Forgot Password" link for password recovery.

3. Main Dashboard: Navigation menu with options for deforestation predictor, trend analyzer, and user profile. Map of Para with selectable points for the deforestation predictor. Options to select future year and view prediction results.

4. Deforestation Predictor: Selected point displayed on the map. Input field or dropdown for selecting future year. Predicted deforestation value displayed prominently.

5. Trend Analyzer: Dropdown for selecting country (India, Zambia, Indonesia, Brazil). Input fields for selecting year gap (e.g., 2012 to 2020). Bar graph displaying deforestation trends over the selected period.

6. User Profile: Display user information and account settings. Option to logout from the account.

7. Overall Design Considerations: Use a color scheme that reflects the environmental theme (greens, browns, earth tones). Ensure the interface is intuitive and easy to navigate. Use icons and visual elements to enhance usability.

5.4 Database Design

In our website's architecture, we implemented a secure and efficient password and username database storage system using SQLite, a lightweight and self-contained relational database management system. This database stores user credentials securely, ensuring confidentiality and integrity of sensitive information.

To achieve this, we designed a table within the SQLite database specifically dedicated to storing user login credentials. Each row in this table represents a user account and contains fields for the username and the hashed password. Hashing algorithms are utilized to convert plaintext passwords into irreversible hashed values before storage. This adds an additional layer of security, preventing unauthorized access to user passwords even in the event of a data breach.

Furthermore, the SQLite database is accessed and managed through secure protocols, such as HTTPS, to encrypt data transmission between the web server and client browser. This mitigates the risk of data interception and ensures the confidentiality of user credentials during transmission.

In addition to security measures, the SQLite database provides scalability and reliability for our website's storage needs. Its self-contained nature simplifies deployment and maintenance, making it an ideal choice for small to medium-scale web applications like ours.

Overall, by implementing a password and username database storage system using SQLite, we ensure that user credentials are securely stored and managed, adhering to best practices in web security and data protection. This robust database solution forms a crucial component of our website's architecture, safeguarding user privacy and facilitating seamless authentication processes for our users.

5.5 Description of Implementation Strategies

The various implementation stratagies used in our project are as follows:

1. Google Earth Engine for Satellite Imagery: We utilized the Google Earth Engine Python API to access and process satellite imagery for deforestation analysis. The API provides a wide range of functionalities for working with geospatial data, making it suitable for our project's needs.

2. Random Forest Algorithm for Deforestation Prediction: We implemented the Random Forest algorithm using the scikit-learn library in Python. This algorithm was chosen for its ability to handle large datasets and its high accuracy in predicting deforestation.

3. Flask for Web Application Development: We used the Flask framework to develop the web application interface for ForestEye. Flask provides a lightweight and flexible

framework for building web applications in Python.

4. MySQL for Database Management: We utilized MySQL for storing user information such as usernames and encrypted passwords. MySQL provides a reliable and scalable database solution for web applications.

In this chapter, we embarked on a comprehensive journey towards designing an integrated system tailored for the deployment and utilization of machine learning algorithms. We began by meticulously identifying pertinent datasets, ensuring their relevance and reliability as the cornerstone for subsequent model training and evaluation. Drawing upon a synthesis of established techniques and innovative approaches, our methodology aimed to develop robust machine learning models capable of accurate prediction, classification, or clustering. Our attention then turned to user interface design, where intuitive layout and informative visualizations empowered users to interact effectively with machine learning models and interpret results. Simultaneously, we laid the foundation for robust data management and model integration through meticulous database design, ensuring data integrity, scalability, and performance. Finally, implementation strategies were elucidated to facilitate a smooth transition from prototype to production, culminating in a functional system poised to harness the transformative power of machine learning across various domains and applications. Through the convergence of dataset identification, methodology formulation, user interface design, database architecture, and implementation strategies, this chapter serves as a roadmap towards the realization of an integrated system primed to address real-world challenges and drive innovation through machine learning.

Chapter 6

Results and Discussions

In this chapter, we unveil the culmination of our machine learning project focused on deforestation prediction in the state of Para, Brazil. We present the end results derived from our predictive models, offering insights into their effectiveness in forecasting deforestation patterns within this critical region. Through quantitative analysis, we delve into numerical metrics such as prediction accuracy, precision, recall, and F1-score, providing a comprehensive evaluation of our models' performance specifically tailored to the unique environmental dynamics of Para. Rigorous testing procedures are conducted to assess the robustness and generalization capabilities of our predictive models, ensuring their reliability across different geographic areas and temporal scales within the state. Additionally, graphical analyses are employed to visualize deforestation trends over time, enabling stakeholders to grasp the magnitude and spatial distribution of forest loss in Para. By integrating quantitative insights, testing procedures, and graphical analyses, this chapter offers a holistic understanding of our machine learning project for deforestation prediction in the state of Para, Brazil, and highlights its significance in informing targeted conservation efforts and sustainable land management strategies in the region.

6.1 Overview

Integrating a machine learning project into a website enables users to predict forest cover in specified coordinates within the state of Pará, Brazil, using the Random Forest Regressor algorithm. Additionally, the website offers a comprehensive trend analysis of forest cover loss in key regions, including India, Brazil, Zambia, and Indonesia. Through graphical representations and statistical analyses, users can visualize historical trends, identify patterns of deforestation, and assess the extent of environmental degradation over time. This combined predictive capability and trend analysis empower stakeholders to make informed decisions, prioritize conservation efforts, and address pressing environmental challenges in these critical regions.

6.2 Testing

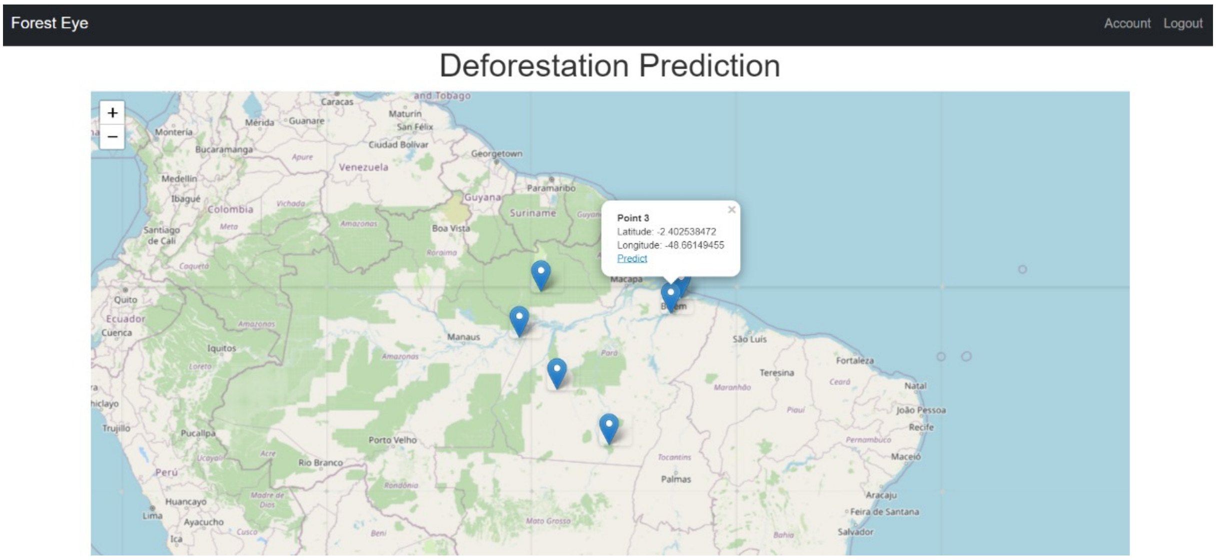


Figure 6.1: The Map

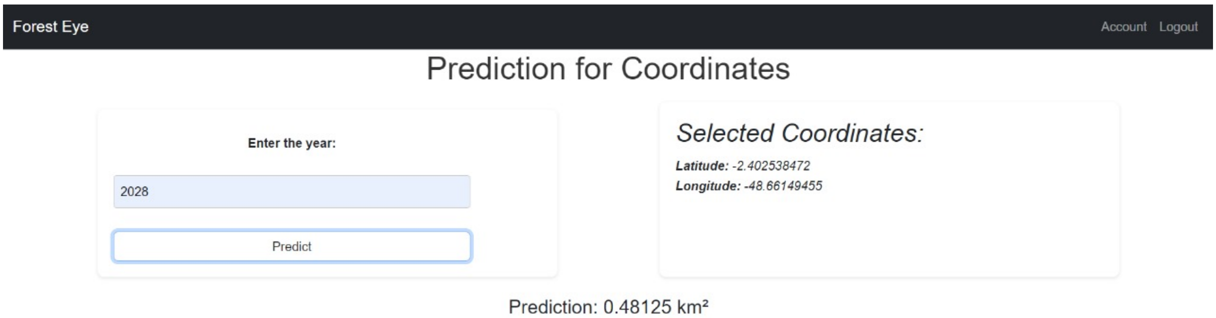


Figure 6.2: Prediction Page

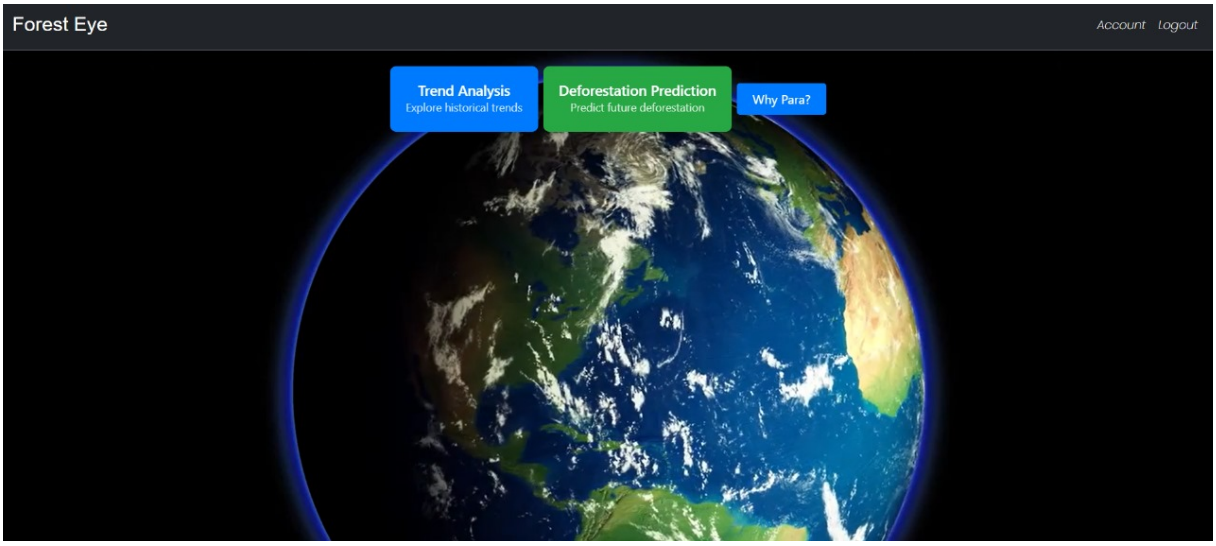


Figure 6.3: Home Page

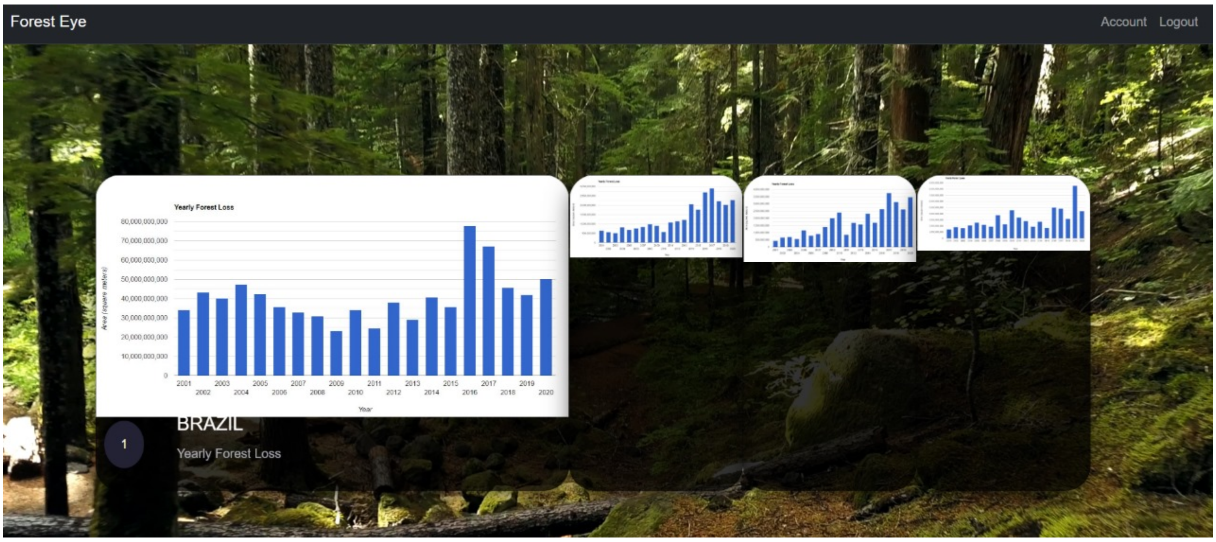


Figure 6.4: Trend Analyzer

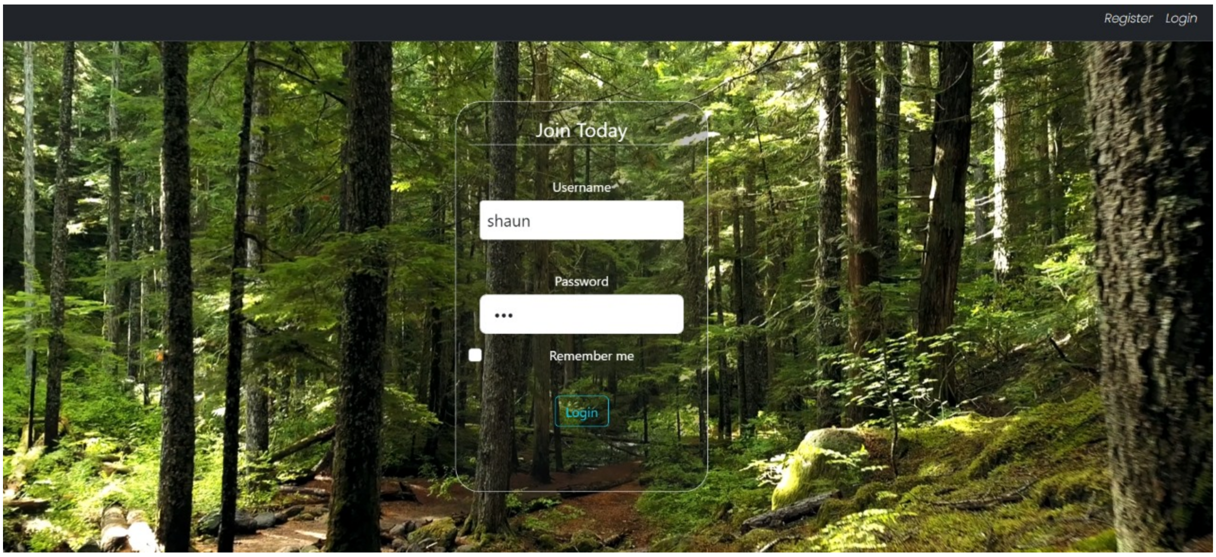


Figure 6.5: Login Page

6.3 Quantitative Results

Model	RMSE Value	MSE Value	MAE Value
XGBoost	0.1490	0.0222	0.0584
Random Forest	0.1445	0.0208	0.0157
LSTM	0.1563	0.0244	0.0783
Linear Regression	0.1624	0.0264	0.1128
SVR	0.1562	0.0244	0.0842

Table 6.1: Comparison between different models

In machine learning, the root mean square error (RMSE) is a common metric used to measure the performance of regression models. It quantifies the average difference between the predicted values and the actual values of the target variable. When comparing different machine learning algorithms, a lower RMSE is generally considered preferable for several reasons.

Firstly, a lower RMSE indicates that the model's predictions are closer to the true values of the target variable. This implies that the model has learned to capture the underlying patterns and relationships in the data more accurately, leading to more reliable predictions. In practical terms, a model with lower RMSE is more likely to provide accurate forecasts or estimates, which is crucial in applications where precision is paramount, such as financial forecasting or medical diagnosis.

Secondly, a lower RMSE suggests that the model has better generalization performance. Generalization refers to the ability of a model to perform well on unseen data, beyond the data it was trained on. A model with lower RMSE is less likely to overfit the training data, meaning it is less susceptible to capturing noise or irrelevant patterns in the data. Instead, it focuses on learning the underlying relationships that generalize well to new data, resulting in more robust and reliable predictions.

Additionally, from a practical standpoint, a lower RMSE may translate to tangible benefits such as cost savings or improved efficiency. For example, in a business context, accurate predictions obtained from a model with lower RMSE could lead to better decision-making, optimized resource allocation, and ultimately, improved profitability.

Overall, machine learning algorithms with lower root mean square error are considered preferable because they provide more accurate, reliable, and generalizable predictions, which are essential for effective decision-making and problem-solving in various domains.

6.4 Graphical Analysis

6.4.1 Comparison of Yearly Forest Loss Area in India, Zambia, Brazil, and Indonesia

The graph illustrates the yearly forest loss area in India, Zambia, Brazil, and Indonesia over a specified time period. Brazil exhibits the highest forest loss among the four countries, making it a significant focal point for deforestation analysis. The x-axis represents different years, while the y-axis indicates the corresponding forest loss area in hectares. This comparison underscores the severity of deforestation in Brazil compared to other countries, influencing its selection as the primary focus of our study.

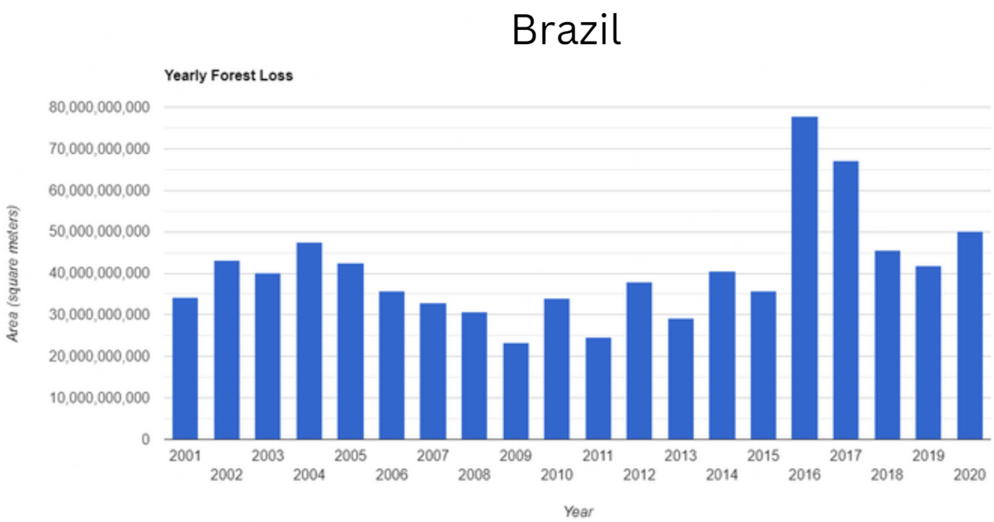


Figure 6.6: Yearly Forest Loss-Brazil

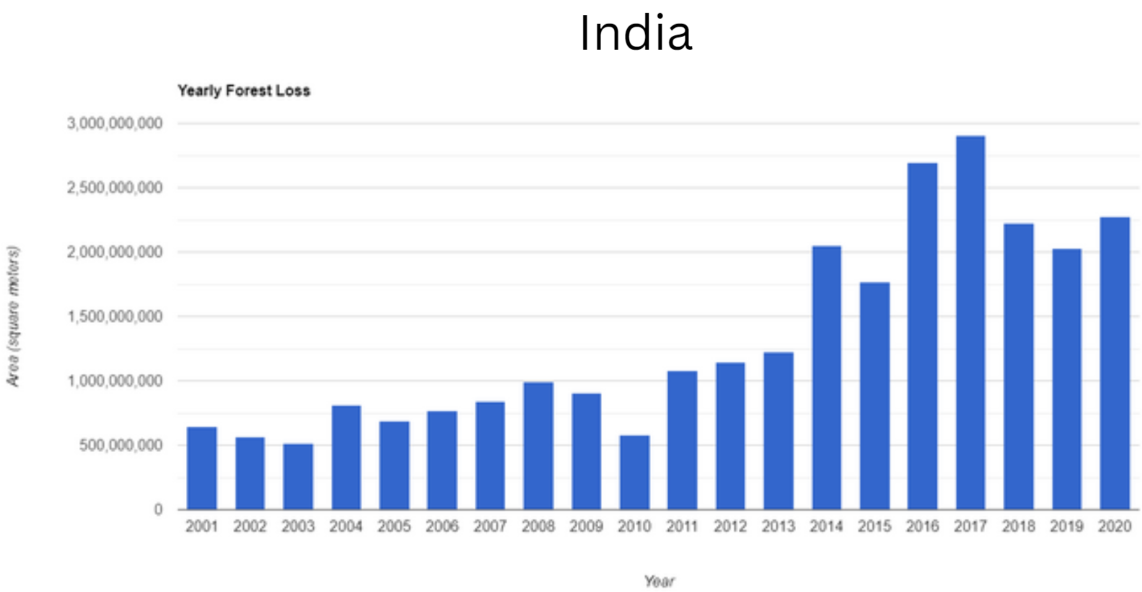


Figure 6.7: Yearly Forest Loss-India

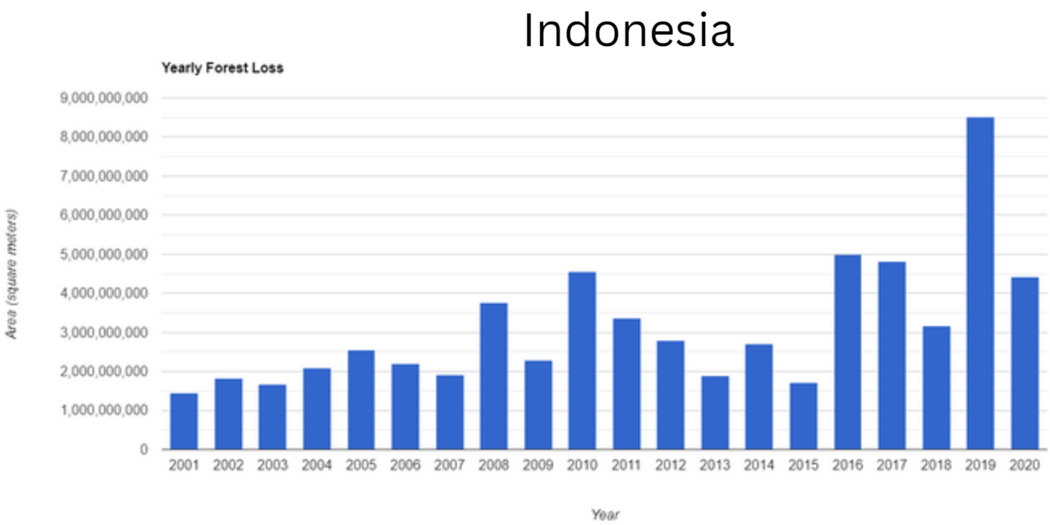


Figure 6.8: Yearly Forest Loss-Indonesia

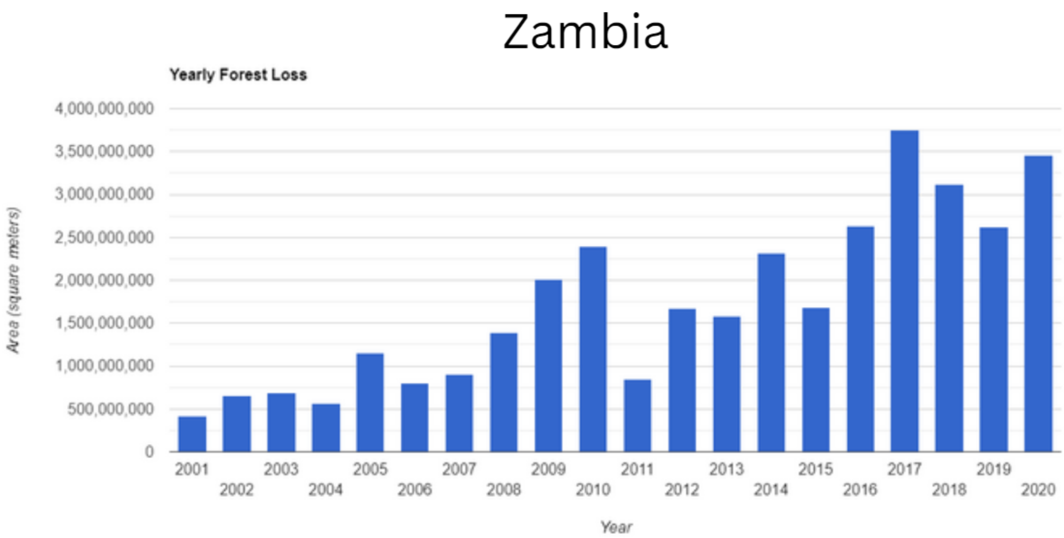


Figure 6.9: Yearly Forest Loss-Zambia

6.5 Discussion

The results of our project, ForestEye, demonstrate the effectiveness of using the Random Forest algorithm for predicting deforestation trends in Para, Brazil. The deforestation predictor module accurately forecasts future deforestation in square kilometers for selected areas based on historical data. Similarly, the trend analyzer module provides insightful visualizations of historical deforestation patterns, aiding in understanding the trends over time.

The choice of the Random Forest algorithm proved to be successful, as it outperformed other machine learning algorithms such as LSTM, Linear Regression, XGBoost, and SVR in terms of RMSE value. This highlights the algorithm's suitability for analyzing and predicting complex deforestation patterns.

One potential deviation in the results could be attributed to the quality and resolution of the satellite imagery used. Higher resolution imagery could provide more detailed and accurate predictions, especially for smaller areas or regions with dense forest cover. Additionally, incorporating more features related to socio-economic factors and land use changes could further improve the accuracy of the predictions.

Overall, the results of ForestEye demonstrate its potential as a valuable tool for monitoring and predicting deforestation trends, contributing to more informed decision-making and effective conservation efforts in Para, Brazil.

In conclusion, the results of ForestEye showcase the effectiveness of the Random Forest algorithm in predicting deforestation trends in Para, Brazil. The application's deforestation predictor accurately forecasts future deforestation, while the trend analyzer provides insightful visualizations of historical patterns. The choice of Random Forest over other algorithms was validated by its superior performance. However, improvements in satellite imagery quality and inclusion of additional features could further enhance prediction accuracy. Overall, ForestEye demonstrates significant potential as a tool for informed decision-making and conservation efforts in deforestation monitoring.

Chapter 7

Conclusions & Future Scope

ForestEye, our web application for deforestation trend analysis and prediction in Para, Brazil, has been successfully implemented. Leveraging the Random Forest algorithm, ForestEye provides users with valuable insights into future deforestation and historical trends. Through a user-friendly interface and interactive visualizations, ForestEye empowers stakeholders to make informed decisions regarding deforestation monitoring and conservation efforts. The use of HTML, CSS, Python, Flask, and MySQL has enabled us to create a robust and efficient system for analyzing deforestation trends.

In the future, ForestEye could be expanded to include more advanced machine learning algorithms for deforestation prediction, such as deep learning models. Additionally, integrating real-time satellite data could enhance the accuracy of deforestation predictions. Furthermore, incorporating social and economic data could provide a more holistic view of deforestation drivers. Finally, expanding the geographical scope of the application to cover other regions facing deforestation challenges would increase its impact and utility on a global scale.

References

[1] J. Ding, N. Xue, G.-S. Xia, X. Bai, W. Yang, and M. Y. Yang, "Object detection in aerial images: A large-scale benchmark and challenges," 2019.

[2] H. Ouchra, A. Belangour, and A. Erraissi, "Machine learning algorithms for satellite image classification using google earth engine and landsat satellite data: Morocco case study," 2018.

[3] E. Kalinicheva, D. Ienco, J. Sublime, and M. Trocan, "Unsupervised change detection analysis in satellite image time series using deep learning combined with graph based approaches," 2020.

[4] W. Li, K. Chen, H. Chen, and Z. Shi, "Geographical knowledge-driven representation learning for remote sensing images," 2019.

[5] Y. Yang, D. Yang, X. Wang, Z. Zhang, and Z. Nawaz, "Testing accuracy of land cover classification algorithms in the qilian mountains based on gee cloud platform," *Remote Sensing*, vol. 13, p. 5064, 2021.

[6] E. M. Sellami and H. Rhinane, "A new approach for mapping land use / land cover using google earth engine: A comparison of composition images," 2019.

List of Publications

1. Monitoring and assessing forest degradation in tropical moist forests for REDD+ Authors: G. P. Asner, J. Mascaro Published in: PLoS ONE, 2014.

2. Predicting deforestation in the Brazilian Amazon using information theory and machine learning" Authors: A. M. Lechner, G. Brown, B. F. Zaitchik Published in: Environmental Modelling Software, 2010.

3. A review of remote sensing of forest biomass and bioenergy: Challenges and opportunities" Authors: M. Wulder, B. Bater, N. C. Coops Published in: Remote Sensing of Environment, 2012.

4. Remote sensing of forest biodiversity" Authors: N. Pettorelli, J. O. Vik, A. Mysterud, et al. Published in: Remote Sensing in Ecology and Conservation, 2014.

Appendix A: Presentation

Final Project Presentation

Deforestation Prediction And Analysis

Members

Team Members:
Richard Sherlin
Roy Rajesh
Shaun Tojan
Tijin T Babu
Guide:
Mr. Harikrishnan M

Problem Definition

- The state of Para in Brazil faces significant challenges due to deforestation, driven by various factors such as agricultural expansion, forest fire and infrastructure development causing severe environmental, social and economic consequences.

Project Objectives

- To create a user friendly platform using machine learning to predict future deforestation in state of Para and analyze historical patterns of different countries.

Novelty of Idea and Scope of Implementation

- Project focuses on State of Para, Brazil.
- Creating dataset from Landsat satellites images
- Urban Planning and Biodiversity conservation.
- Act as an early warning system for stakeholders, alerting them to areas at risk of deforestation.
- Serve as an educational resource for students, researchers, and educators interested in deforestation.

Literature Review

PAPER	METHODOLOGY	ADVANTAGE	DISADVANTAGE
Object Detection in Aerial Images: A Large-Scale Benchmark and Challenges Jian Ding , Nan Xue , Gui-Song Xia , Xiang Bai , Wen Yang , Michael Ying Yang	Dataset Expansion,Code Library creation and Algorithm Evaluation	- Large-scale dataset - Code library and website	-Limited generalizability - Limited scope
Machine Learning Algorithms for Satellite Image Classification Using Google Earth Engine and Landsat Satellite Data: Morocco Case Study HAFSA OUCHRA ABDESSAMAD BELANGOUR1 AND ALLAE ERRAISSI2	GEE platform and machine learning algorithms	-Use of advanced machine learning algorithms -Accurate land cover information	-Limited ground truth data -Limited spatial resolution

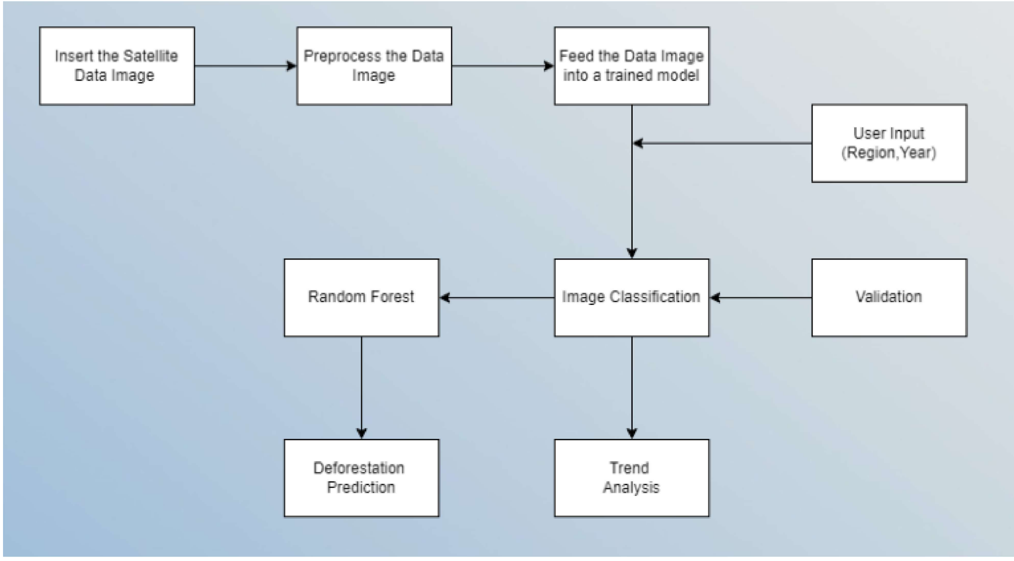
Contd.

PAPER	METHODOLOGY	ADVANTAGE	DISADVANTAGE
Unsupervised Change Detection Analysis in Satellite Image Time Series using deep learning combined with Graph based approaches Ekaterina Kalinicheva , Dino Ienco , Jérémie Sublime, and Maria Trocan	End-to-End approach for change detection and clustering for satellite image time series ,which combines deep learning with graph based approaches	-Reference data for model training is not required -the method is able to identify nontrivial changes	-Difficult to implement as it requires complex deep-learning algorithms
Geographical Knowledge-Driven Representation Learning for Remote Sensing Images Wenyuan Li , Keyan Chen , Hao Chen , and Zhenwei Shi , Member, IEEE	The proposes method, GeoKR, utilizes geographical knowledge to extract supervision information and construct an efficient representation learning framework for removing the influence of noise labels.	-Geographical knowledge to improve network performance. - Reducing the demand for annotated data and outperforming existing pretraining methods.	-Limited generalizability -Difficult to implement

Methodology

- Data Preparation: Collection of satellite images of coordinates for Para from 2008 to 2024,collected from Google Earth Engine. The image is then classified using the random forest classifier. Then the area is calculated based on classified image.Then using the area the a dataset is created.
- Model Training: Training the multiple models like Random Forest, XGBoost, Linear Regression, SVR and LSTM on the created dataset with hyperparameters and evaluating its performance using metrics like RMSE,MSE and MAE.
- Module Implementation: Implementing the deforestation predictor and trend analyzer modules, allowing users to select coordinates and future years for prediction, as well as choose year gaps and countries for trend analysis.Predictions are based on the trained Random Forest model.

Architecture Diagram



Architecture Diagram

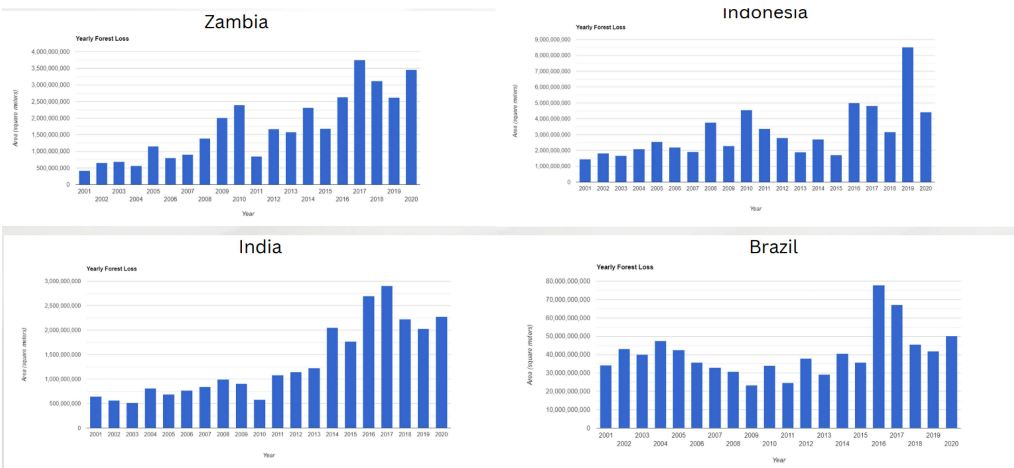
Results - 100% Evaluation

- Identified the highly deforested area
- Analyzed the deforested areas of different countries using Google Earth Engine.
- Collected dataset for different latitude and longitude in Para over a period of 2008 to 2024 using Google Earth Engine and other online resources.
- Preprocessing of the dataset
- Trained the dataset using different models like XGBoost, Random Forest, Linear Regression, SVR and LSTM.

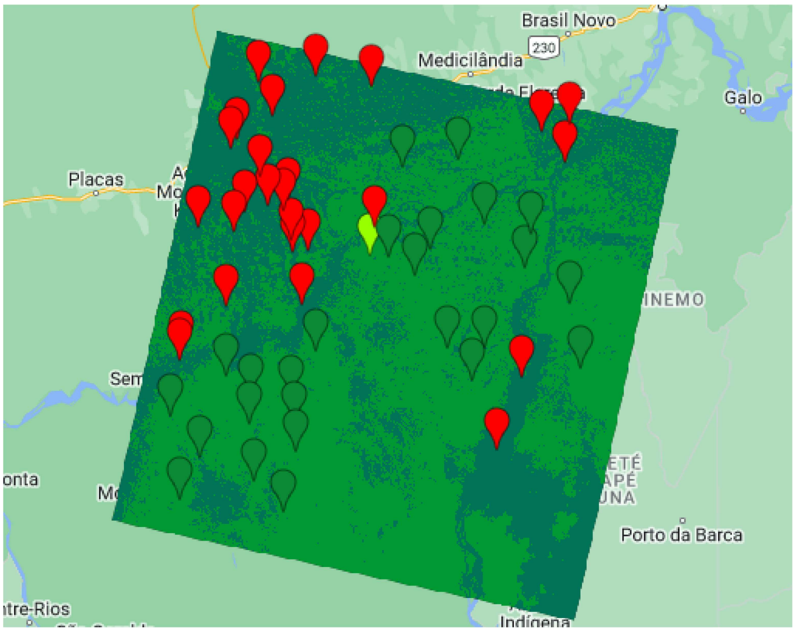
Contd.

- Created a flask web application
- The user can select points from the map and input year to predict the forest cover.
- The user can understand the trend of deforestation for different countries.

Contd.



Contd.



Contd.

Model	RMSE Value	MSE Value	MAE Value
XGBoost	0.1490	0.0222	0.0584
Random Forest	0.1445	0.0208	0.0157
LSTM	0.1563	0.0244	0.0783
Linear Regression	0.1624	0.0264	0.1128
SVR	0.1562	0.0244	0.0842

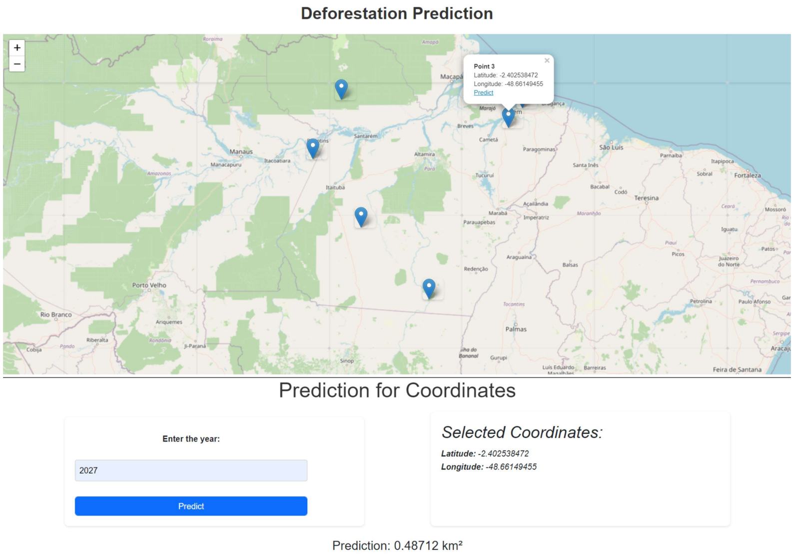
Results

Contd.

Deforestation Dashboard
Trend Analysis
Explore historical trends
Deforestation Prediction
Predict future deforestation
Why Para?

Yearly Forest Loss
INDIA
Yearly Forest Loss

Contd.



Deforestation Prediction
Point 3
Latitude: -2.402538472
Longitude: -48.66149455
Predict
Manaus
Porto Velho
Rio Branco
Palmas
Teresina
Fortaleza
São Luís
Prediction for Coordinates
Enter the year:
2027
Predict
Selected Coordinates:
Latitude: -2.402538472
Longitude: -48.66149455
Prediction: 0.48712 km²

Future Scope

- The project's framework could be applied to other regions, broadening its impact on global deforestation efforts.
- Integrate climate change adaptation strategies into the platform, considering how deforestation affects regional climate patterns.
- Integrate soil composition and terrain data to understand how these factors influence deforestation.

Task Distribution

- Dataset Collection- Richard, Roy, Shaun, Tijin.
- Random Forest Classifier- Richard
- Frontend -Shaun, Roy, Tijin
- Flask -Richard, Tijin
- Map - Shaun, Roy
- Model Training - Richard, Roy, Shaun, Tijin

Conclusion

- Our project leverages Landsat satellite imagery to create a robust dataset. Through machine learning, it predicts future forest areas, aiding conservation decisions and raising awareness for sustainable land management. It also shows the deforestation trends of different countries during past years.

References

- Object Detection in Aerial Images: A Large-Scale Benchmark and Challenges Jian Ding , Nan Xue , Gui-Song Xia , Xiang Bai , Wen Yang , Michael Ying Yang
- Machine Learning Algorithms for Satellite Image Classification Using Google Earth Engine and Landsat Satellite Data: Morocco Case Study HAFSA OUCHRA ABDESSAMAD BELANGOUR1 AND ALLAE ERRAISSI2
- Unsupervised Change Detection Analysis in Satellite Image Time Series using deep learning combined with Graph based approaches Ekaterina Kalinicheva , Dino Ienco , Jérémie Sublime, and Maria Trocan
- Geographical Knowledge-Driven Representation Learning for Remote Sensing Images Wenyuan Li , Keyan Chen , Hao Chen , and Zhenwei Shi , Member, IEEE

Status Of Paper Publication

- Conference name:2024 International Conference on Signal Processing, Computation, Electronics, Power, and Telecommunication
- Submission date: 25/04/2024
- Notification Of Acceptance: 01/05/2024
- Registration Starts: 02/05/2024
- Registration Deadline: 31/05/2024
- Conference Date: July 4-5, 2024

Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes

Vision, Mission, Programme Outcomes and Course Outcomes

Institute Vision

To evolve into a premier technological institution, moulding eminent professionals with creative minds, innovative ideas and sound practical skill, and to shape a future where technology works for the enrichment of mankind.

Institute Mission

To impart state-of-the-art knowledge to individuals in various technological disciplines and to inculcate in them a high degree of social consciousness and human values, thereby enabling them to face the challenges of life with courage and conviction.

Department Vision

To become a centre of excellence in Computer Science and Engineering, moulding professionals catering to the research and professional needs of national and international organizations.

Department Mission

To inspire and nurture students, with up-to-date knowledge in Computer Science and Engineering, ethics, team spirit, leadership abilities, innovation and creativity to come out with solutions meeting societal needs.

Programme Outcomes (PO)

Engineering Graduates will be able to:

1. Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and Team work: Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

10. Communication: Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective reports documentation. Make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Programme Specific Outcomes (PSO)

A graduate of the Computer Science and Engineering Program will demonstrate:

PSO1: Computer Science Specific Skills
The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas by understanding the core principles and concepts of computer science and thereby engage in national grand challenges.

PSO2: Programming and Software Development Skills
The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry.

PSO3: Professional Skills
The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur.

Course Outcomes (CO)

Course Outcome 1: Model and solve real world problems by applying knowledge across domains (Cognitive knowledge level: Apply).

Course Outcome 2: Develop products, processes or technologies for sustainable and socially relevant applications (Cognitive knowledge level: Apply).

Course Outcome 3: Function effectively as an individual and as a leader in diverse teams and to comprehend and execute designated tasks (Cognitive knowledge level: Apply).

Course Outcome 4: Plan and execute tasks utilizing available resources within timelines, following ethical and professional norms (Cognitive knowledge level: Apply).

Course Outcome 5: Identify technology/research gaps and propose innovative/creative solutions (Cognitive knowledge level: Analyze).

Course Outcome 6: Organize and communicate technical and scientific findings effectively in written and oral forms (Cognitive knowledge level: Apply).

Appendix C: CO-PO-PSO Mapping

CO-PO AND CO-PSO MAPPING

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO 1	2	2	2	1	2	2	2	1	1	1	1	2	3		
CO 2	2	2	2		1	3	3	1	1		1	1		2	
CO 3									3	2	2	1			3
CO 4					2			3	2	2	3	2			3
CO 5	2	3	3	1	2							1	3		
CO 6					2			2	2	3	1	1			3

3/2/1: high/medium/low

JUSTIFICATIONS FOR CO-PO MAPPING

MAPPING	LOW/MEDIUM/ HIGH	JUSTIFICATION
100003/ CS722U.1-PO1	M	Knowledge in the area of technology for project development using various tools results in better modeling.
100003/ CS722U.1-PO2	M	Knowledge acquired in the selected area of project development can be used to identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions.

100003/ CS722U.1-PO3	M	Can use the acquired knowledge in designing solutions to complex problems.
100003/ CS722U.1-PO4	M	Can use the acquired knowledge in designing solutions to complex problems.
100003/ CS722U.1-PO5	H	Students are able to interpret, improve and redefine technical aspects for design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
100003/ CS722U.1-PO6	M	Students are able to interpret, improve and redefine technical aspects by applying contextual knowledge to assess societal, health and consequential responsibilities relevant to professional engineering practices.
100003/ CS722U.1-PO7	M	Project development based on societal and environmental context solution identification is the need for sustainable development.
100003/ CS722U.1-PO8	L	Project development should be based on professional ethics and responsibilities.
100003/ CS722U.1-PO9	L	Project development using a systematic approach based on well defined principles will result in teamwork.
100003/ CS722U.1-PO10	M	Project brings technological changes in society.

100003/ CS722U.1-PO11	H	Acquiring knowledge for project development gathers skills in design, analysis, development and implementation of algorithms.
100003/ CS722U.1-PO12	H	Knowledge for project development contributes engineering skills in computing & information gatherings.
100003/ CS722U.2-PO1	H	Knowledge acquired for project development will also include systematic planning, developing, testing and implementation in computer science solutions in various domains.
100003/ CS722U.2-PO2	H	Project design and development using a systematic approach brings knowledge in mathematics and engineering fundamentals.
100003/ CS722U.2-PO3	H	Identifying, formulating and analyzing the project results in a systematic approach.
100003/ CS722U.2-PO5	H	Systematic approach is the tip for solving complex problems in various domains.
100003/ CS722U.2-PO6	H	Systematic approach in the technical and design aspects provide valid conclusions.
100003/ CS722U.2-PO7	H	Systematic approach in the technical and design aspects demonstrate the knowledge of sustainable development.

100003/ CS722U.2-PO8	M	Identification and justification of technical aspects of project development demonstrates the need for sustainable development.
100003/ CS722U.2-PO9	H	Apply professional ethics and responsibilities in engineering practice of development.
100003/ CS722U.2-PO11	H	Systematic approach also includes effective reporting and documentation which gives clear instructions.
100003/ CS722U.2-PO12	M	Project development using a systematic approach based on well defined principles will result in better teamwork.
100003/ CS722U.3-PO9	H	Project development as a team brings the ability to engage in independent and lifelong learning.
100003/ CS722U.3-PO10	H	Identification, formulation and justification in technical aspects will be based on acquiring skills in design and development of algorithms.
100003/ CS722U.3-PO11	H	Identification, formulation and justification in technical aspects provides the betterment of life in various domains.
100003/ CS722U.3-PO12	H	Students are able to interpret, improve and redefine technical aspects with mathematics, science and engineering fundamentals for the solutions of complex problems.

100003/ CS722U.4- PO5	H	Students are able to interpret, improve and redefine technical aspects with identification formulation and analysis of complex problems.
100003/ CS722U.4- PO8	H	Students are able to interpret, improve and redefine technical aspects to meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
100003/ CS722U.4- PO9	H	Students are able to interpret, improve and redefine technical aspects for design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
100003/ CS722U.4- PO10	H	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools for better products.
100003/ CS722U.4- PO11	M	Students are able to interpret, improve and redefine technical aspects by applying contextual knowledge to assess societal, health and consequential responsibilities relevant to professional engineering practices.
100003/ CS722U.4- PO12	H	Students are able to interpret, improve and redefine technical aspects for demonstrating the knowledge of, and need for sustainable development.
100003/ CS722U.5- PO1	H	Students are able to interpret, improve and redefine technical aspects, apply ethical principles and commit to

		professional ethics and responsibilities and norms of the engineering practice.
100003/ CS722U.5-PO2	M	Students are able to interpret, improve and redefine technical aspects, communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
100003/ CS722U.5-PO3	H	Students are able to interpret, improve and redefine technical aspects to demonstrate knowledge and understanding of the engineering and management principle in multidisciplinary environments.
100003/ CS722U.5-PO4	H	Students are able to interpret, improve and redefine technical aspects, recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.
100003/ CS722U.5-PO5	M	Students are able to interpret, improve and redefine technical aspects in acquiring skills to design, analyze and develop algorithms and implement those using high-level programming languages.
100003/ CS722U.5-PO12	M	Students are able to interpret, improve and redefine technical aspects and contribute their engineering skills in computing and information engineering domains like

		network design and administration, database design and knowledge engineering.
100003/ CS722U.6-PO5	M	Students are able to interpret, improve and redefine technical aspects and develop strong skills in systematic planning, developing, testing, implementing and providing IT solutions for different domains which helps in the betterment of life.
100003/ CS722U.6-PO8	H	Students will be able to associate with a team as an effective team player for the development of technical projects by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
100003/ CS722U.6-PO9	H	Students will be able to associate with a team as an effective team player to Identify, formulate, review research literature, and analyze complex engineering problems
100003/ CS722U.6-PO10	M	Students will be able to associate with a team as an effective team player for designing solutions to complex engineering problems and design system components.
100003/ CS722U.6-PO11	M	Students will be able to associate with a team as an effective team player, use research-based knowledge and research methods including design of experiments, analysis and interpretation of data.

100003/ CS722U.6-PO12	H	Students will be able to associate with a team as an effective team player, applying ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
100003/ CS722U.1-PSO1	H	Students are able to develop Computer Science Specific Skills by modeling and solving problems.
100003/ CS722U.2-PSO2	M	Developing products, processes or technologies for sustainable and socially relevant applications can promote Programming and Software Development Skills.
100003/ CS722U.3-PSO3	H	Working in a team can result in the effective development of Professional Skills.
100003/ CS722U.4-PSO3	H	Planning and scheduling can result in the effective development of Professional Skills.
100003/ CS722U.5-PSO1	H	Students are able to develop Computer Science Specific Skills by creating innovative solutions to problems.
100003/ CS722U.6-PSO3	H	Organizing and communicating technical and scientific findings can help in the effective development of Professional Skills.