TREES AND BUILDINGS EXTRACTION FROM SATELLITE IMAGES USING DEEP NEURAL NETWORK

A PROJECT REPORT

Submitted by

SREE KRISHNA B

(2023176030)

A report for the phase-I of the project
submitted to the Faculty of

INFORMATION AND COMMUNICATION ENGINEERING

in partial fulfillment
for the award of the degree
of

MASTER OF TECHNOLOGY
in

INFORMATION TECHNOLOGY

SPECIALIZATION IN AI & DS

DEPARTMENT OF INFORMATION SCIENCE AND TECHNOLOGY
COLLEGE OF ENGINEERING, GUINDY
ANNA UNIVERSITY
CHENNAI 600 025

DECEMBER 2024

ANNA UNIVERSITY

CHENNAI - 600 025

BONA FIDE CERTIFICATE

Certified that this project report titled TREES AND BUILDINGS EXTRACTION FROM SATELLITE IMAGE USING DEEP NEURAL NETWORK is the bona fide work of Sree Krishna B (2023176030) who carried out project work under my supervision. Certified further that to the best of my knowledge and belief, the work reported herein does not form part of any other thesis or dissertation on the basis of which a degree or an award was conferred on an earlier occasion on this or any other candidate.

PLACE:CHENNAI
DATE:

Dr.K. VANI
PROFESSOR
PROJECT GUIDE
DEPARTMENT OF IST, CEG
ANNA UNIVERSITY
CHENNAI 600025

COUNTERSIGNED

DR. S. SWAMYNATHAN
HEAD OF THE DEPARTMENT
DEPARTMENT OF INFORMATION SCIENCE AND TECHNOLOGY
COLLEGE OF ENGINEERING, GUINDY
ANNA UNIVERSITY
CHENNAI 600025

ABSTRACT

Conventional methods for identifying trees and structures in satellite data rely on manual annotation or traditional image processing, which lack accuracy and scalability. These approaches struggle with complex textures, overlapping features, and uneven lighting, limiting their effectiveness. They are time-consuming, error-prone, and unsuitable for real-time applications like infrastructure planning or urban map updates, hindering efficient urban management.

This project aims to address these limitations by employing deep learning-based semantic segmentation models—U-Net and SegNet. U-Net captures fine spatial details with its encoder-decoder structure and SegNet ensures computational efficiency for real-time segmentation. The models are trained on multispectral satellite imagery, combining RGB and NIR bands for enhanced vegetation detection and urban feature classification, with the Adam optimizer ensuring robust segmentation performance.

The outcome of the project highlights the successful application of deep learning models to automate urban map updates by classifying features such as trees and buildings. This system provides valuable insights for key applications, including infrastructure planning, environmental monitoring, and disaster management, enabling more info.

Phase 2 will refine the models to address complex urban scenarios, introduce real-time map updates, and enhance vegetation monitoring. Real-world validations will ensure practical and sustainable urban development.

ACKNOWLEDGEMENT

I wish to express our deep sense of gratitude and sincere thanks to our project supervisor, DR. K. VANI, Professor, Department of Information Science and Technology, College of Engineering, Guindy, for her invaluable guidance, continuous support, and encouragement throughout the course of this project. Her insightful feedback and constant motivation were instrumental in shaping the outcome of this work.

We are also deeply indebted to DR. S. SWAMYNATHAN, Professor and Head of the Department of Information Science and Technology, Anna University, Chennai, for providing the necessary resources and facilities that enabled us to undertake and complete this project.

We extend our heartfelt thanks to the project committee members, DR. S. SRIDHAR, Professor, DR. G. GEETHA , Assistant professor, and DR. D. NARASHIMAN, Teaching Fellow, for their constructive feedback and valuable suggestions that enriched our work.We also thank all the faculty and non-teaching staff members of the Information Science and Technology, CEG Campus, Anna University, Chennai, for their valuable support throughout the course of my project work.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

BMA	Bayesian Model Averaging
CAUM	Context-Aware Upsampling Module
CNNs	Convolutional Neural Networks
GAFS	Genetic Algorithms for Feature Selection
GCAM	Global Context Aggregation
KDE	Kernel Density Estimation
NRMSE	Normalized Root Mean Squared Errors
NIR	Near-Infrared bands
POI	Point of Interest
RANSAC	Random sample consensus
RGB	Red, Green, Blue
SAM	Segment Anything Model
ViTs	Vision Transformers
YOLO	You Only Look Once

CHAPTER 1

INTRODUCTION

A crucial aspect of urban planning and environmental monitoring is the ability to accurately identify and classify urban features such as trees and buildings. Conventional methods for urban analysis often rely on traditional approaches, including manual annotation, rule-based image processing techniques, and outdated mapping processes. While these methods have been used in the past with some success, they pose significant challenges.

Manual annotation and traditional processing demand substantial human effort, making it difficult to manage large datasets efficiently, particularly for extensive or rapidly changing urban areas. This resource-intensive workflow is prone to errors, inconsistencies, and oversights, leading to unreliable results that can hinder effective decision-making. Static methods also fail to adapt to the dynamic nature of urban landscapes, including evolving structures, changing vegetation patterns, and diverse environmental conditions.Furthermore, traditional systems struggle to process large-scale urban datasets, making them impractical for applications such as city-wide infrastructure planning or environmental monitoring. Additionally, conventional techniques cannot process or integrate real-time satellite imagery, limiting their capacity to provide timely insights for urgent needs like disaster management or urban map updates.

1.1 BACKGROUND

Urban planning, environmental monitoring, disaster management, and resource optimization are just a few of the applications that depend on the precise classification of urban features like buildings and trees. There is an increasing need for accurate and current mapping to facilitate well-informed decision-making as urban areas continue to grow quickly. Monitoring and analyzing large-scale urban environments effectively is made possible by satellite imagery, especially multispectral imagery. Differentiating between natural and artificial features is made possible by satellite imagery, which records information across a variety of spectral bands, including the visible (RGB) and non-visible Near-Infrared (NIR). Buildings, which display unique visual traits in the RGB bands, and trees, which reflect strongly in the NIR band, require this capability to be recognized.

Traditional approaches for classifying urban features are still popular despite the development of remote sensing technologies, but they have many drawbacks. Manual procedures are very time-consuming and require a great deal of skill, such as visual interpretation of satellite images and simple image processing methods. These techniques are subject to human error, which can produce inconsistent and inaccurate results. Additionally, manual methods are unable to handle the complexity of urban environments, where it is challenging to distinguish between buildings and trees due to overlapping features, spectral similarities, and texture variations. Traditional methods are therefore unsustainable and ineffective for contemporary urban analysis, particularly when used in expansive or dynamic urban areas.

The need for automated, scalable, and precise solutions for urban feature extraction is what drives this project's efforts to address these issues. Recent developments in deep learning have demonstrated enormous promise for using semantic segmentation techniques to increase the precision and effectiveness of classification tasks. The limitations of manual methods can be overcome by using deep learning models and multispectral satellite imagery to classify urban features at the pixel level with high precision. For the analysis of intricate, sizable datasets, this automation guarantees scalability, minimizes errors, and cuts down on human labor. Deep learning's application to the classification of urban features has great potential to aid in disaster relief efforts, improve environmental monitoring, and support urban planning.

1.2 PROBLEM STATEMENT

In urban analysis, precisely identifying and extracting urban features—like buildings and trees—from multispectral satellite images is still a major challenge. Rapid and accurate mapping of these features is essential for a number of uses, such as environmental monitoring and urban planning, as urbanization increases. The demands of efficiently analyzing large and complex datasets cannot be met by conventional methods like manual interpretation and simple image processing techniques. There is an obvious need for automated methods that can produce accurate and dependable results as urban environments continue to grow.

This project addresses the problem of identifying and extracting trees and buildings using advanced deep learning methods. By employing Ensemble Convolutional Neural Networks (CNNs), the study aims to classify and extract these urban features at the pixel level, utilizing the potential of multispectral satellite imagery. Multispectral data, particularly RGB and Near-Infrared (NIR) bands, provide additional spectral information that aids in distinguishing between vegetation and man-made structures. However, efficiently processing such data to classify complex urban features accurately remains a major problem.

In order to produce insightful and thorough results that can be immediately used to update urban street maps, this study also incorporates semantic classification techniques. The effectiveness of conventional approaches is limited by their inability to adjust to dynamic variables and changing urban landscapes. The reliability of urban data analysis is eventually increased by addressing these constraints through automated, deep learning-based techniques that enable scalable and precise urban feature extraction.

1.3 OBJECTIVE

The objective of this project is to create an automated system that uses deep learning to classify and extract buildings and trees from multispectral satellite imagery. Because of their intricate layouts and overlapping features, urban areas necessitate precise and effective methods for mapping and tracking urban elements. The project intends to overcome the drawbacks of conventional manual and simple image processing techniques by utilizing contemporary deep learning techniques, offering a scalable and dependable solution for urban feature extraction.

The project focuses on putting advanced semantic segmentation models like SegNet and U-Net into practice in order to accomplish this. These models have demonstrated efficacy in pixel-level classification tasks, where SegNet provides computational efficiency and U-Net is well-suited for maintaining spatial details. Multispectral data, such as RGB (Red, Green, Blue) and NIR bands, can be processed by the segmentation models. By examining their spectral reflectance characteristics, the NIR band is essential for detecting vegetation like trees, whereas the RGB bands are crucial for recognizing man-made structures like buildings.

An ensemble regression-based approach is incorporated into the project to improve the classification process's accuracy and resilience. By utilizing each model's unique strengths, this approach integrates the results of several models to enhance overall performance. The difficulties presented by intricate and varied urban datasets are addressed by ensemble approaches, which guarantee increased generalization and dependability. In order to provide ground-truth data for performance evaluation and guarantee consistency in results, the models are trained, validated, and tested on annotated multispectral datasets.

As a result, this system can be used to create current urban maps, which are necessary for resource allocation, urban planning, and environmental monitoring. Furthermore, by providing precise and timely data for estimating damage and organizing recovery efforts, the extracted information can support disaster response efforts. In addition to lowering the need for manual intervention, the project's automation of urban feature extraction lays the groundwork for scalable, effective, and precise mapping solutions that meet the demands of contemporary urban analysis.

1.4 ORGANIZATION OF CHAPTERS

Chapter 1: Introduction The project's background, problem statement, and motivation are presented in this chapter. It also explains the importance of the work and lays out the project's goals and objectives.

Chapter 2: Literature Review This chapter examines the literature on deep learning-based techniques, remote sensing, and urban feature extraction. It talks about the drawbacks of conventional methods and emphasizes the value of models such as SegNet and U-Net.

Chapter 3: Methodology The methodology chapter describes the dataset, preprocessing steps, and the implementation of deep learning models for semantic segmentation. It also details the ensemble regression approach and the evaluation process.

Chapter 4: Implementaion This chapter explains the steps taken to implement the segmentation models, including data preprocessing, training U-Net and SegNet, and evaluating performance. It highlights the metrics used, such as IoU and Precision, and describes the validation and visualization processes..

Chapter 5: Result and Discussion This chapter presents the experimental results, comparing U-Net and SegNet in segmenting urban features. It discusses model performance through metrics like IoU and F1-Score, along with visual results, addressing challenges and practical implications.

Chapter 6: Conclusion and Future work This chapter summarizes the findings, emphasizing the success of U-Net and SegNet in urban segmentation. Future work aims to expand segmentation to include more urban features, improve accuracy with advanced architectures, and enable real-world validation for comprehensive map updates.

CHAPTER 2

LITERATURE SURVEY

A literature survey was done by surveying research papers. The limitations and the knowledge gained from the papers will help us to create a better system

2.1 TREE MAPPING

Recent advancements in urban tree analysis leverage deep learning models and multispectral imagery to significantly enhance the detection and classification of urban trees. Tools such as convolutional neural networks (CNNs) and high-resolution multispectral imaging systems are rigorously evaluated for their effectiveness in addressing urban forestry challenges[1]. Each technology offers distinct benefits tailored to meet the specific demands of urban forestry projects, ensuring that the methodologies applied are both efficient and effective. Integrating CNNs with multispectral imagery aids in the precise identification and categorization of tree species over large urban landscapes, greatly improving the accuracy and scalability of urban forest inventories. Furthermore, the deployment of deep-learning models, particularly encoder-decoder networks combined with LiDAR data, has been effective in creating detailed 3D visualizations of urban tree canopies[2]. This powerful combination not only boosts classification accuracy but also facilitates dynamic interaction within these 3D environments, enhancing urban landscape management and planning activities.

2.2 BUILDING DETECTION

Recent advancements in urban planning leverage machine learning segmentation techniques to enhance urban development. Semantic segmentation, using high-resolution satellite imagery and architectures like UNet and UNet++, effectively categorizes various building types, from industrial complexes to residential villas[3]. This is crucial for urban planners in mapping, zoning, and resource management by accurately identifying building distributions and characteristics.

Advanced Mask R-CNN integrated with PointRend is also used to delineate individual buildings in densely populated areas[4]. This precise segmentation is critical for population density assessments, disaster response, and infrastructure development, especially where buildings are closely spaced.

A deep learning framework is designed for extracting building footprints from high-resolution aerial imagery, utilizing an advanced encoder-decoder architecture[5]. The encoder captures multi-scale, global features with densely connected convolutional and transition blocks, while the decoder employs deconvolution layers to reconstruct detailed segmentation maps. This method addresses the challenges of varied building sizes and complex urban layouts, enhancing the accuracy of identifying buildings within satellite images. Trained end-to-end with a hybrid loss function, the framework demonstrates significant improvements in precision, recall, and Intersection over Union (IoU) metrics over existing methods, making it a more efficient tool for urban planning and disaster management.

2.3 ADVANCED ENVIRONMENTAL MONITORING

Advanced deep learning technologies applied to environmental monitoring and urban planning demonstrate transformative impacts. By integrating Vision Transformers (ViTs) with WorldView-3 satellite data, one study achieves precise mapping and segmentation of date palm trees, showing significant improvements in mapping accuracy essential for creating detailed agricultural inventories and enhancing geospatial databases[6]. Concurrently, another study employs the Segment Anything Model (SAM) with vegetation indices to accurately map urban tree crowns from Google Earth imagery, emphasizing the vital role of urban forestry in carbon sequestration and its potential as a nature-based solution to climate change[7]. Together, these studies underscore how cutting-edge computational methods can effectively address complex environmental challenges, providing robust tools for ecosystem management and sustainable urban development. They highlight the capacity of these technologies to advance scientific understanding and contribute practical solutions for climate mitigation and urban planning.

2.4 URBAN BUILDING ANALYSIS

Innovations in urban building analysis have utilized a combination of deep learning and remote sensing techniques to address diverse urban planning challenges. One study integrates remote sensing imagery with Point of Interest (POI) data to develop a stepwise framework for identifying urban building functions. By employing spatial similarity and Kernel Density Estimation (KDE), this approach enhances the identification of building functions, achieving greater accuracy and completeness, particularly in complex urban environments like Wuhan, China[8] . Another study focuses on the detection of rooftop thermal bridges using aerial thermal imagery captured by drones. It evaluates deep learning architectures, including Swin Transformers and MaskRCNN, to accurately identify thermal anomalies on rooftops[9]. This work emphasizes the role of drone-acquired thermal data and advanced computational models in improving energy efficiency assessments and guiding retrofitting plans for urban districts Together, these studies demonstrate the potential of integrating remote sensing data with advanced analytical models to solve critical urban and environmental challenges.

2.5 AUTOMATED TREE MAPPING

Deep learning techniques and remote sensing have been effectively utilized to advance urban tree analysis and management. One approach leverages models such as YOLO and Faster R-CNN to detect and geolocate urban trees using a combination of aerial, satellite, and ground-level imagery. By employing triangulation methods with data from Google Street View, this method achieves high positional accuracy, overcoming challenges like canopy occlusion and varying resolutions[10]. It demonstrates scalability for urban forest inventories with a detection accuracy of 79% and a positional error of 60 cm.

Another approach integrates Google Street View images with algorithms like YOLOv3 and Panoptic-DeepLab to perform detailed structural analysis of urban trees[11]. This includes identifying tree species and estimating tree profiles such as diameter, height, and crown width. The model combines object detection and semantic segmentation, achieving a mean average precision (mAP) of 0.564 for species classification and normalized root mean squared errors (NRMSE) of 0.24 and 0.44 for height and diameter estimation, respectively.

These methods underscore the transformative role of deep learning in automating urban tree detection, classification, and structural analysis. They offer scalable and efficient solutions for monitoring urban greenery, significantly contributing to sustainable urban planning and ecosystem management.

2.6 FORESTRY DEEP LEARNING

TrunkNet, a multiscale attention-based deep learning method, is designed for detecting and segmenting tree trunks in urban settings. Utilizing the Salient-Trunk Dataset (ST-D) with pixel-level annotations, it incorporates texture attention and multiscale information fusion mechanisms to emphasize textural features and overcome occlusion challenges caused by foliage, shadows, or urban clutter[12]. This approach achieves high accuracy in identifying trunks across diverse species and urban scenes, including partially occluded or complex structures, supporting tasks such as tree pruning, species classification, and urban ecosystem monitoring. Similarly, an adjusted Mask R-CNN framework is applied to detect and segment standing dead trees in dense mixed forests using low-resolution CIR aerial imagery[13]. By employing transfer learning and data augmentation, this method achieves a mean average precision (mAP) of 0.85, recall of 0.88, and an F1 score of 0.87, providing an automated approach to delineating dead tree crowns. This framework is critical for biodiversity conservation, forest health monitoring, and carbon storage assessments in challenging dense forest environments. Together, these methods demonstrate the transformative role of deep learning in forestry by offering scalable and efficient solutions for tree detection, segmentation, and management across diverse landscapes while addressing the complexities of real-world applications.

2.7 SEMANTIC SEGMENTATION FOR URBAN POINT CLOUDS

Advancements in semantic segmentation techniques for 3D urban point clouds have enabled precise extraction and classification of urban features. One approach introduces the Context-Aware Network (CAN)[14], specifically designed to process large-scale point clouds. By leveraging modules like Local Feature Aggregation (LFAM) and Global Context Aggregation (GCAM), this method captures detailed geometric features and long-range dependencies, ensuring accurate segmentation even in complex urban environments. Additionally, the Context-Aware Upsampling Module (CAUM) refines features effectively, balancing low-level and high-level details. Tested on datasets such as Semantic3D and Tongji-3D, this method demonstrates high accuracy and robustness, highlighting its potential for urban scene understanding and city modeling.

Another methodology combines RANSAC and Hough Transform (HT) techniques to detect and segment primitive shapes such as planes, cylinders, and spheres within large-scale point clouds. This approach excels in partitioning urban elements like roofs, walls, and pavements into meaningful components, even under noisy and occluded conditions[15]. By focusing on primitive recognition and geometric feature extraction, this method supports critical applications such as building reconstruction and digital twin creation. Evaluations on the Vaihingen dataset and similar benchmarks highlight its precision and computational efficiency, making it a valuable tool for processing urban data and enabling scalable urban management solutions[16].

2.8 BURN SEVERITY ANALYSIS

The integration of advanced machine learning techniques and satellite imagery has significantly enhanced the prediction and management of forest fires. One approach focuses on predicting forest fire risks by analyzing climate change patterns using temperature records and greenhouse gas emission data[17]. By combining traditional machine learning methods, such as ARIMA and logistic regression, with deep learning models, this method improves the accuracy of temperature and rainfall forecasts, enabling better predictions of forest fire risks. The methodology incorporates exploratory data analysis, normalization, and feature selection techniques like Genetic Algorithms for Feature Selection (GAFS) to optimize the prediction process. This approach demonstrates the effectiveness of leveraging historical data and advanced computational techniques to support proactive resource allocation and disaster management.

The use of Sentinel-2 satellite imagery to estimate burnt areas and classify burn severity offers another dimension to forest fire management[18]. By applying Bayesian Model Averaging (BMA) techniques and oversampling methods like SMOTE to address imbalanced datasets, machine learning models such as CART and ExtraTrees Regressor achieve high performance. The BMA model, in particular, reached an accuracy of 96.35%, enabling precise mapping of burnt areas and classification of fire severity into high, medium, and low categories. This capability is invaluable for identifying regions requiring immediate intervention, supporting post-fire recovery efforts, and contributing to the development of early warning systems. Together, these methods underscore the transformative potential of combining satellite data with advanced analytical techniques to address the challenges posed by forest fires effectively.

2.9 SUMMARY OF THE EXISTING WORK

The literature survey highlights advancements in deep learning and remote sensing technologies that address challenges in urban analysis, environmental monitoring, and disaster management. Techniques such as Context-Aware Networks (CAN) and encoder-decoder architectures like U-Net have been applied to urban data, enabling precise classification of urban features like roofs, walls, and pavements. These methodologies support applications like city modeling, digital twin creation, and infrastructure management. Advanced forestry models, including TrunkNet and Mask R-CNN frameworks, have been utilized for tasks like tree trunk segmentation and standing dead tree detection, contributing to biodiversity conservation and forest health monitoring. Additionally, Vision Transformers (ViTs) and the Segment Anything Model (SAM) have further enhanced segmentation accuracy for tree crown mapping and carbon sequestration studies. These techniques demonstrate the transformative potential of deep learning in providing scalable solutions to complex urban and environmental challenges.

Despite these advancements, several challenges persist. Existing methods often fail to address real-world complexities, such as noise and occlusion in urban point cloud data or the dynamic variability of forest environments during fire events. Models for urban analysis struggle with scalability when processing large-scale datasets and lack integration of detailed behavioral or environmental factors, such as changes in building materials or urban layouts over time. In forestry applications, segmentation models often overlook nuances like tree species variability and the effects of occlusion by dense canopies. Fire risk models and burn severity estimations frequently neglect factors such as wind patterns and the flammability of specific vegetation types, limiting their predictive accuracy. These gaps highlight the need for more comprehensive solutions that integrate dynamic, real-time data to address the complexities of urban and environmental systems effectively.

CHAPTER 3

SYSTEM ARCHITECTURE

This chapter focuses on the system design of the proposed urban feature extraction system for satellite imagery analysis. It presents the overall architecture and provides an overview of how various components interact to facilitate efficient extraction and classification of urban features such as trees and buildings. The design emphasizes the integration of deep learning models, multispectral data, and preprocessing techniques to enhance accuracy and scalability. Furthermore, this chapter outlines the data acquisition process, methodologies employed, and the integration of advanced segmentation techniques. A detailed explanation of each module is provided, including data preprocessing, spectral analysis, model implementation, feature segmentation, and output generation for urban planning and environmental monitoring.

3.1 PROPOSED ARCHITECTURE

This section outlines the design of an advanced urban feature extraction system aimed at enhancing urban planning and environmental monitoring. The system processes high-resolution satellite imagery to extract and classify trees and buildings while addressing challenges such as overlapping features, spectral similarities, and complex textures. It integrates deep learning models for semantic segmentation, multispectral data analysis for feature differentiation, and classification techniques for accurate feature mapping. The system leverages RGB and NIR bands to improve the accuracy of segmentation, ensuring reliable detection and classification of urban features.

The process starts with input satellite imagery, which is preprocessed for alignment and noise reduction. Datasets, including Sentinel-2 imagery and annotated ground truth datasets, are utilized to train models. Semantic segmentation is performed using models like U-Net and SegNet, followed by classification to distinguish trees and buildings. The results are validated using metrics such as IoU and F1-score to ensure precision and reliability. The final outputs, including segmented and classified maps, are visualized through an interactive interface and stored in a cloud-based system for accessibility. This system provides actionable insights, offering a scalable solution for urban feature extraction and supporting critical applications in infrastructure development, disaster management, and resource optimization.

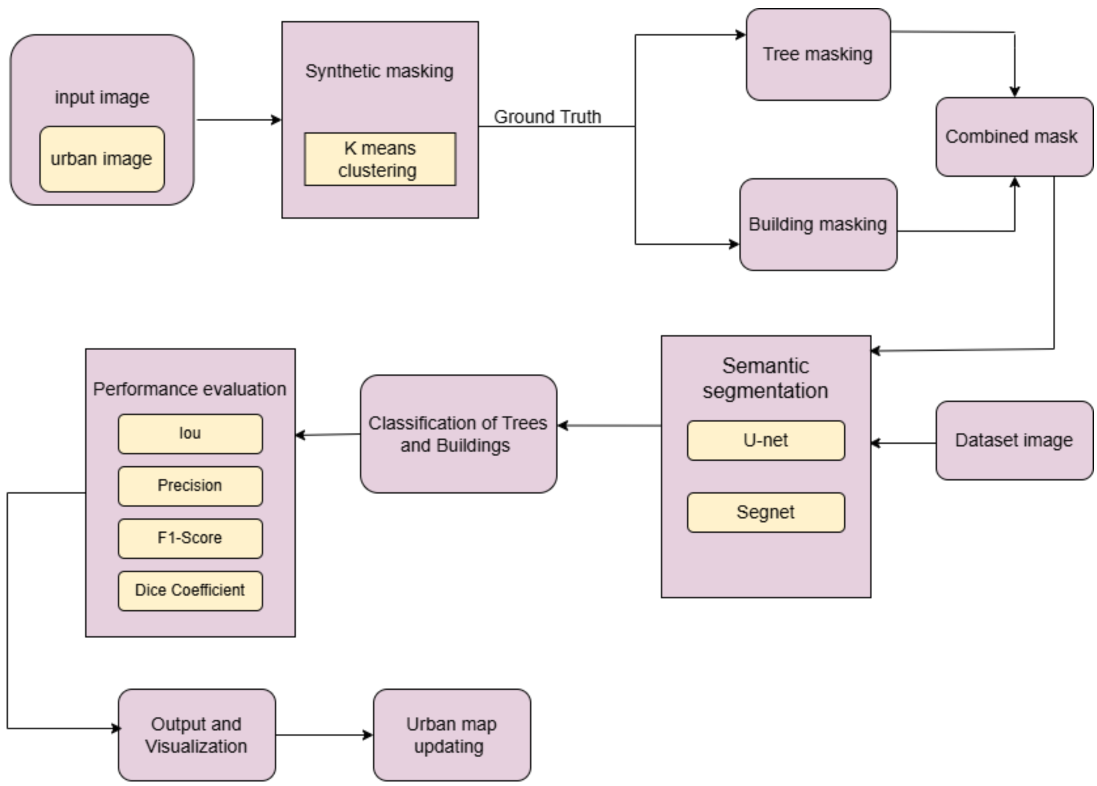

Figure 3.1: Urban Detection architecture Diagram

3.2 PROPOSED SYSTEM ARCHITECTURE

The proposed system for urban feature extraction employs a tightly integrated architecture, which is detailed in the architecture diagram shown in Figure 3.1.High-resolution satellite imagery, including RGB and Near-Infrared (NIR) bands, is used as the input to the system, providing essential spatial and spectral information for accurate feature extraction. Ground truth masks are generated using K-Means Clustering, which partitions the input image into distinct clusters representing trees, buildings, and background. These masks are utilized for training segmentation models such as U-Net and SegNet, which segment the input images precisely at the pixel level to identify and delineate trees and buildings. Following segmentation, the extracted features are classified into trees and buildings based on their characteristics and spectral properties. The system's accuracy and reliability are assessed using performance metrics such as IoU, precision, and F1-score to ensure robust and consistent results. The final segmented and classified results are visualized using an interactive interface, providing clear and interpretable outputs that help users analyze and validate the results effectively. Using the final outputs, the system updates urban maps by integrating the segmented and classified features into a comprehensive map, which includes details of tree distribution and building footprints, enabling improved urban planning, infrastructure management, and environmental monitoring.

3.3 MODEL DESCRPTION

The description of the modules and the respective figures are mentioned below.

3.3.1 Synthetic Mask

Synthetic mask generation is performed using K-Means Clustering, an unsupervised machine learning technique, for the segmentation of urban images. The process begins with a high-resolution input image, typically a satellite image that contains urban features like trees, buildings, and background. The input image serves as the foundation for further processing.

To generate the masks, the K-Means Clustering algorithm is applied, where pixels in the input image are grouped into clusters based on their spectral and spatial characteristics. This step effectively identifies distinct regions such as buildings, trees, and open spaces by analyzing pixel intensities and spectral similarities. The clustering process helps in isolating meaningful segments from the image, forming synthetic masks that serve as ground truth data.

The generated masks act as labeled data, enabling precise identification and classification of urban features, forming the foundation for further tasks like feature classification, performance evaluation, and urban map updating.

3.3.2 Ground Truth Generation

High-resolution photos of cities are first loaded into the system through the 'Input Image' module. 'Synthetic Masking' is applied to these images using K-means clustering as seen in Figure 3.2, which divides the image into segments according to pixel values in order to distinguish between urban features like buildings, roads, and vegetation. This helps with more in-depth analysis by generating a preliminary rough division of the image.

Then the system employs 'Tree Masking' and 'Building Masking.'

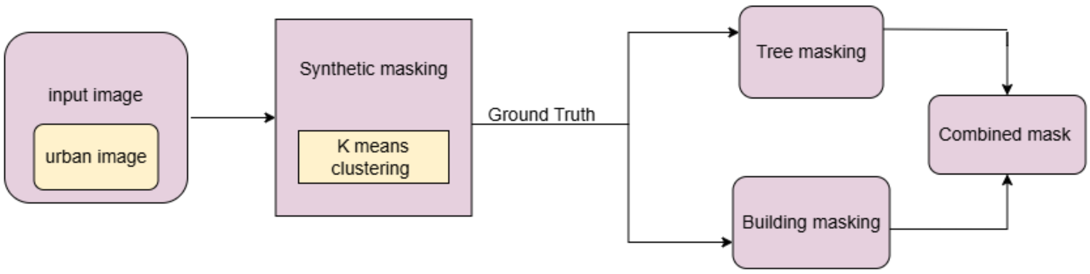

Figure 3.2: Mask Generation

Tree Masking, which is crucial for monitoring urban green spaces, applies specific algorithms to the segments in order to precisely identify and mask areas covered by trees. In contrast, Building Masking highlights buildings using techniques that support infrastructure studies and urban planning.

The segmented features of buildings and trees are integrated into a final mask after the two masks have been created. Deep learning model training requires this combined mask, which acts as the input image's ground truth representation. These models accurately segment and classify urban features using the labeled data, allowing for additional analysis for environmental monitoring and urban planning.

3.4 SEMANTIC SEGMENTATION

The Semantic Segmentation module analyzes urban images to identify and categorize distinct elements like buildings and trees into separate, meaningful groups based on pixel characteristics.

3.4.1 U-Net

Utilizing its strong architecture to provide high precision and detail,

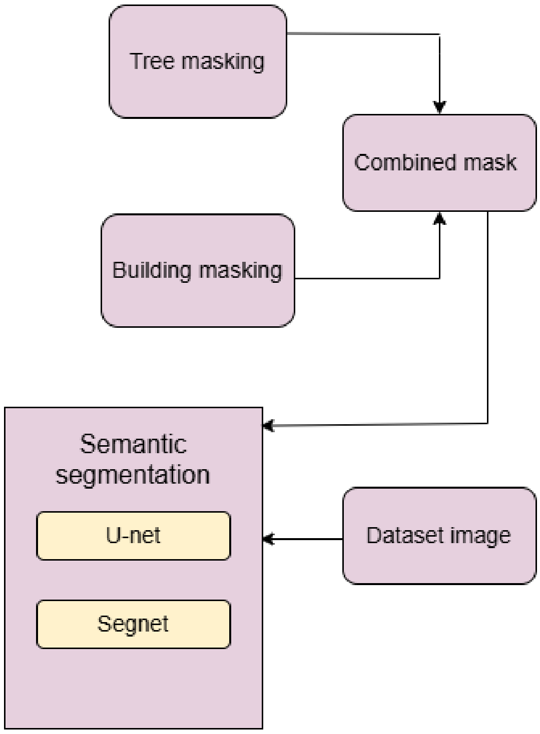

Figure 3.3: Image Segmentation

the U-net model plays a crucial part in your project's urban image segmentation. Because of its remarkable capacity to manage intricate spatial details in scenes with a high population density, this model—which was initially created for medical image segmentation—has been modified for use in urban settings. The architecture of the U-net is a significant advantage; its "U" shape is made up of an expanding path that guarantees accurate localization of different urban elements like roads, buildings, and vegetation, and a contracting path that gathers contextual information.

The contracting path helps the model encode the crucial contextual information required for precise segmentation by gradually decreasing the image's spatial dimension while deepening feature maps. The expanding path, on the other hand, maintains precise and in-depth localization throughout the image by reconstructing the segmentation maps from the encoded features. This technique enables the U-net to distinguish between closely spaced features, which is a crucial capability for in-depth urban analysis.

Using the U-net in this project improves the precision of identifying and separating distinct features in the urban environment, which is essential for later tasks like building and tree masking. Because of these features, U-net is a very useful tool for environmental monitoring and urban planning, guaranteeing accurate and trustworthy segmentation results for subsequent analysis and decision-making.

3.4.2 Seg-Net

Segnet is used in this project because of its strong segmentation capabilities, especially for tasks involving large amounts of images. The deep encoder-decoder architecture used in this model is very efficient at processing intricate urban landscapes. In order to effectively capture the crucial details required for precise segmentation, Segnet's encoder component compresses the input images into a lower-dimensional feature space. This is especially helpful for defining different urban elements such as streets, structures, and green areas.

The decoder element of the Segnet model is essential because it restores the segmentation maps from the compressed features to their original image resolution. The production of comprehensible and practical segmentation outputs depends on pixel-wise accuracy, which is ensured by this function. Segnet is very helpful for in-depth environmental analysis and urban planning because it can accurately classify every pixel.

Additionally, Segnet's operational efficiency—which uses less memory and computational power than other deep learning models—is especially helpful when managing the big datasets that are common in urban imaging projects. It is an essential part of the project methodology because of its high fidelity feature segmentation capabilities, which enable complex analyses like planning infrastructure projects or assessing urban density.

3.5 CLASSIFICATION

The ”Classification of Trees and Buildings” stage of urban landscape analysis is an intricate procedure that expands upon the results of semantic segmentation models and converts them into useful information for environmental monitoring and urban planning. In order to distinguish between man-made structures like buildings and natural vegetation, especially trees, which are vital components of urban ecosystems, this stage is necessary.

3.5.1 Foundations of the Classification Process

Semantic segmentation is the first step in classification from Figure 3.4, where sophisticated models like Segnet and U-net produce intricate segmentation maps. These maps use pixel properties to visually separate the urban landscape into different areas. The primary characteristics of each segmented area—such as texture, color, and form—that correlate to either the natural or constructed environments are used to assign preliminary labels. The following classification tasks are built upon this segmentation.

3.5.2 Advanced Techniques in Feature Classification

After segmentation is finished, particular classification algorithms are applied, carefully examining the segmented areas to ascertain whether they are buildings or trees. This conclusion is not simple and is largely dependent on machine learning methods that can analyze intricate data patterns. Here, algorithms such as K-means clustering. This algorithms have learned the

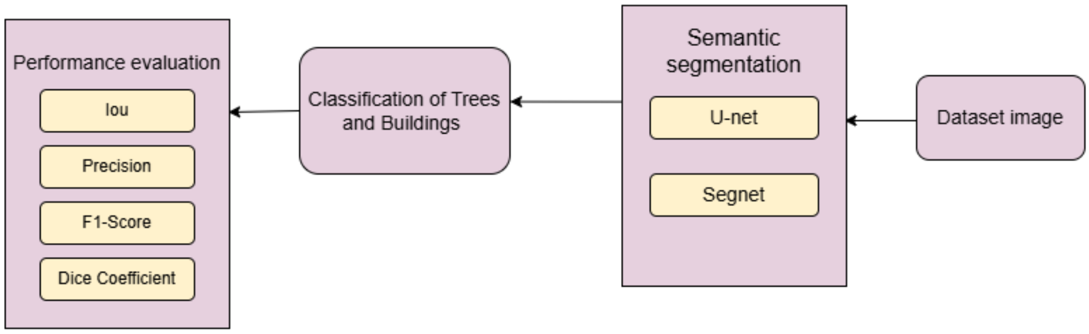

Figure 3.4: Urban Classification

distinctive features connected to trees and buildings by being trained on a variety of datasets.

By analyzing different aspects of each segment, such as its size, shape, and relationship to its surroundings, these machine learning models improve the classification's accuracy. A small green space that is encircled by other green spaces in a park-like configuration, is more likely to be categorized as vegetation.

3.5.3 Refinement and Contextual Analysis

Additionally, relational analysis and spatial context are incorporated into the classification process, which greatly improves its accuracy. This entails taking into account each segment separately as well as examining its relationships with neighboring segments. Based on the combined features of nearby regions, this contextual analysis aids in validating or updating preliminary classifications.

In order to increase precision, the process also incorporates iterative refinement steps where the outputs are continuously modified. In order to reduce errors like false positives—which mistakenly identify non-tree areas as trees—and false negatives—which fail to identify actual tree areas—these refinements entail changing the thresholds and parameters within the classification algorithms.

3.6 PERFORMANCE METRICS AND EVALUATION

The ”Performance Evaluation” module depicted in your diagram emphasizes four critical metrics, each instrumental in assessing the accuracy and effectiveness of classification or segmentation models, especially in complex tasks such as urban image segmentation.

3.6.1 Intersection Over Union

Intersection over Union (IoU), also known as the Jaccard Index, is a fundamental measure used to evaluate the similarity and diversity between the predicted segmentation and the actual ground truth in image processing. It quantifies how well the segmentation model predictions overlap with the true data and is calculated using the formula:

$$\text{IoU} = \frac{\text{Area of Overlap}}{\text{Area of Union}} \tag{3.1}$$

This metric is crucial for assessing the spatial accuracy of a model’s predictions, indicating how precisely the segmented areas match the true segments in urban landscapes.

3.6.2 Precision

Precision measures the accuracy of positive predictions made by the model, reflecting the proportion of positive identifications that were actually correct. It is defined as:

$$\text{Precision} = \frac{TP}{TP+FP} \tag{3.2}$$

where TP is true positives and FP is false positives. Precision is particularly important in applications where the cost of a false positive is high, ensuring that the model reliably labels only relevant samples as positive.

3.6.3 F1-Score

The F1-Score provides a balance between precision and recall, making it a critical measure when both aspects are equally important. The F1-Score is especially valuable in datasets with an uneven class distribution and is computed as:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{3.3}$$

This score helps in understanding the model's accuracy in detecting all relevant instances (recall) and the precision of these detections.

3.6.4 Dice Coefficient

The Dice Coefficient, similar to IoU but focusing more explicitly on the number of true positives, measures the model's effectiveness in capturing the true spatial distribution of the classified features. It is calculated by:

$$\text{Dice Coefficient} = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{3.4}$$

where FN is false negatives. The Dice Coefficient is particularly useful in medical image segmentation and other applications where the model must be sensitive to the precise boundaries of small or intricately shaped objects.

Together, these metrics form a robust framework for evaluating the performance of segmentation and classification models. They ensure that models are not only accurate in detecting the correct areas but also precise in how they categorize each segment, which is critical for applications like urban planning where precise and reliable data segmentation leads to better-informed decisions.

3.7 OUTPUT AND VISUALIZATION

The Output and Visualization component plays a crucial role in transforming complex, segmented data into a clear and actionable format that supports urban analysis and decision-making. This stage ensures that results from segmentation models, such as U-net and Segnet, are not only accurate but also easy to interpret and analyze. The process starts with data synthesis, where raw outputs are aggregated and organized into a structured format. The segmented areas, such as trees, buildings, and roads, are processed into labeled coordinates, dimensions (e.g., area, height, and perimeter), and object counts. Additionally, spatial relationships between features are analyzed to identify clusters of green spaces, calculate distances between buildings, or highlight patterns in urban layouts. The resulting synthesized data can be stored in standard geospatial formats such as GeoJSON, Shapefiles, or Raster Layers, making it suitable for visualization and further analysis.

Once the data is synthesized, it is visualized using a variety of techniques to enhance its clarity and usability. Thematic maps are created to highlight specific features, such as green spaces shown in shades of green and built-up areas displayed in gray. These maps provide an easy-to-understand representation of the urban landscape. For more detailed insights, 3D visualizations are used to represent terrain features and buildings, enabling an in-depth understanding of their height, spatial distribution, and structure. Color-coded overlays and heatmaps further assist in identifying areas of high density, such as regions with limited vegetation or intense infrastructure. Interactive dashboards are often integrated into the visualization process, combining visual maps with graphs, charts, and performance metrics, such as total green cover or building-to-tree ratios. These techniques allow users to identify patterns, trends, and anomalies in the urban landscape, facilitating more informed decision-making.

Interactivity is a key aspect of modern visualization platforms, often enhanced through Geographic Information Systems (GIS) tools. These tools allow users to dynamically explore the visualized data, providing options to pan, zoom, and rotate maps for closer inspection. Features can be queried to extract detailed information—for example, selecting a tree cluster might reveal its area, density, and location, while clicking on a building could display its height, type, and proximity to other structures. Layer management features allow users to toggle different urban features on and off, such as vegetation, roads, and water bodies, enabling focused analysis. Additionally, advanced tools support time-series comparisons, allowing users to analyze urban changes over time, such as tracking green space loss or monitoring infrastructure growth.

By synthesizing raw outputs into meaningful visual formats and incorporating interactive tools, this component provides an intuitive and accessible way to interpret results. It allows urban planners, policymakers, and environmental scientists to identify areas needing attention, such as green space deficits, urban sprawl, or infrastructure gaps. It also supports long-term monitoring of urban development, ensuring that resources are allocated efficiently and sustainably. Ultimately, the Output and Visualization process bridges the gap between data analysis and actionable insights, offering a robust platform for making data-driven decisions to improve urban environments.

3.8 URBAN MAP UPDATION

The Urban Map Updating component plays a significant role in ensuring that urban maps reflect the most up-to-date and accurate representation of the current landscape. This process involves integrating the newly processed and visualized data—such as classified trees, buildings, and other urban features—into existing mapping systems. Urban map updating is critical for applications like urban planning, infrastructure development, environmental monitoring, and disaster management. By incorporating real-time or newly acquired data, this component ensures that urban maps remain relevant, reliable, and actionable for stakeholders.

The process begins with data integration, where the outputs from the segmentation and classification stages are merged into Geographic Information Systems (GIS) or other urban mapping platforms. These platforms allow the creation, editing, and visualization of spatial data through layered structures. The segmented outputs—such as buildings and tree cover—are organized into separate layers, which can be updated individually without affecting the entire map structure. For example, if newly classified vegetation areas are added to a green space layer, the updated layer reflects this change seamlessly, enabling precise management of urban greenery. Similarly, changes to built-up areas, such as new constructions or demolitions, are added to the corresponding infrastructure layer.

Once the data integration process is complete, the next step focuses on accuracy improvement. This involves cross-referencing the updated map with other reliable sources, such as satellite imagery, aerial photographs, or on-ground survey data. Spatial inconsistencies, such as misaligned features or outdated regions, are corrected during this step to ensure that the map maintains a high level of precision. Advanced GIS tools often incorporate automated quality checks, which help detect errors or discrepancies and suggest corrections. The improved accuracy enhances the usability of the updated map for detailed urban planning and analysis.

An important aspect of urban map updating is continuous monitoring and dynamic updates. Urban environments are constantly changing due to factors such as population growth, infrastructure expansion, or natural changes like vegetation growth and loss. To keep up with these changes, the map updating process can be performed at regular intervals (e.g., annually or bi-annually) or in near-real-time, depending on the availability of updated data. Continuous monitoring is facilitated through tools like satellite imagery, drone mapping, which provide real-time inputs that can be processed and integrated into urban maps efficiently. This ensures that urban planners and decision-makers always have access to the most current and accurate information.

The updated urban maps offer numerous benefits to stakeholders. They provide a reliable foundation for tasks such as urban planning, where precise information about land use, green spaces, and infrastructure is essential for designing sustainable and livable cities. Similarly, these maps aid in infrastructure management by identifying areas that require maintenance or development. In environmental monitoring, updated maps can help track changes in vegetation cover, identify areas prone to deforestation, or assess the impact of urbanization on natural resources. Additionally, these maps are vital for disaster preparedness and response, as they offer accurate spatial data to plan evacuation routes, emergency response zones, and resource allocation.

CHAPTER 4

IMPLEMENTATION

This section provides a complete, step-by-step breakdown of the implementation process, detailing all the stages of segmentation, classification, and visualization. The workflow focuses on identifying and masking trees and buildings in urban landscape images, generating combined masks, and evaluating the model performance using various metrics. Algorithm 4.1 outlines the complete methodology, starting from data preparation to model evaluation and visualization. Each stage is meticulously designed to ensure efficient processing and accurate urban feature segmentation, while the results are assessed using metrics like IoU, Dice Coefficient, Precision, and Recall to gauge the model's effectiveness. This structured approach ensures clarity and reproducibility of the process.

4.1 DATA PREPARATION

Data preparation is the first step in the implementation process and serves as the basis for the complete analysis. Images of urban environments in high resolution make up the data used in this research. Trees, buildings, highways, and other aspects that are essential to comprehending and evaluating metropolitan landscapes are captured in these photos. Since they directly affect the precision of ensuing segmentation and classification tasks, the data's quality and resolution are essential. Even minute details, like a single tree or a smaller building, may be precisely recognized and processed thanks to high-resolution photos.

To facilitate the workflow, it is important to establish a clear structure

Algorithm 4.1 Overall Algorithm for Urban Feature Segmentation and Classification

Require: Dataset D containing images and corresponding masks, Number of classes $num\_classes$, Model type M (e.g., U-Net, SegNet, Mask R-CNN)
Ensure: Segmentation results and performance metrics

▷ Step 1: Data Preparation
Load the dataset D with images I and ground truth masks M_{gt}
Resize images and masks to a fixed size (H,W)
Normalize pixel values of images to $[0,1]$
Split the dataset into training, validation, and test sets $(X_{train},Y_{train},X_{val},Y_{val},X_{test},Y_{test})$

▷ Step 2: Model Initialization
Select a deep learning model M based on project requirements (e.g., U-Net, SegNet)
Define the input shape (H,W,C) and number of classes $num\_classes$

▷ Step 3: Model Training
Compile the model with:
    Loss function: categorical crossentropy
    Optimizer: Adam optimizer with learning rate lr
    Metrics: Accuracy
Train the model on (X_{train},Y_{train}) using validation data (X_{val},Y_{val})
Save the best model weights based on validation loss

▷ Step 4: Model Evaluation
Load the test set (X_{test},Y_{test})
Evaluate the trained model to compute test loss and accuracy
Predict segmentation masks M_{pred} for test images X_{test}

▷ Step 5: Performance Metrics
Calculate pixel-wise metrics:
    Accuracy, Precision, Recall, F1 Score, IoU, Dice Coefficient
Generate and save a confusion matrix for the predicted and ground truth masks

▷ Step 6: Visualization of Results
for each image i in X_{test} (or first n images) **do**
    Visualize the original image I_i
    Visualize the ground truth mask M_{gt}^{i}
    Visualize the predicted mask M_{pred}^{i}
    Save the visualization for reporting
end for

▷ Step 7: Results and Discussion
Analyze the performance metrics and discuss the strengths and limitations of the models
Compare results of different models (if applicable, e.g., U-Net vs. SegNet)
Highlight potential improvements for future work

for organizing the data and outputs. This involves defining the locations of the input data and creating separate directories for the results. Specific folders are created for different types of segmentation masks, including tree masks, building masks, and combined masks. This structured approach ensures that the outputs from various stages of the process are stored in an orderly manner, making it easy to retrieve and analyze the results at later stages.

Proper data organization is essential not only for maintaining clarity but also for streamlining the process. It allows for systematic analysis and prevents confusion when dealing with multiple datasets and outputs. By separating the results into dedicated directories, users can focus on specific aspects of the analysis, such as evaluating the accuracy of tree segmentation or comparing building detection results. This careful preparation and organization of data lay a strong foundation for the subsequent steps in the implementation process, ensuring that the analysis remains efficient and error-free.

4.2 IDENTIFYING TREES USING NDVI

The identification of trees within urban landscapes is an essential part of the analysis, as trees contribute significantly to ecological balance and urban aesthetics. To detect trees in images, we use the Normalized Difference Vegetation Index (NDVI), a widely adopted technique in remote sensing. NDVI leverages the distinct properties of vegetation to differentiate it from other features in an image. Trees reflect a significant amount of green light due to their chlorophyll content, which is why they appear green to our eyes, while simultaneously absorbing red light for photosynthesis. This unique behavior is the basis of NDVI, allowing us to quantify vegetation cover in an image.

NDVI is calculated for each pixel using the formula:

$$NDVI = \frac{\text{Green} - \text{Red}}{\text{Green} + \text{Red}} \tag{4.1}$$

The result of this calculation is a value ranging from -1 to 1, where higher values represent vegetation (e.g., trees and plants), and lower values indicate non-vegetative elements (e.g., buildings, roads, or water). By applying this formula to the green and red channels of an urban image, we generate an NDVI map that highlights areas covered by vegetation, effectively distinguishing them from man-made structures and other non-vegetative elements.

After calculating the NDVI values, the resulting image may still contain noise, which refers to small, unwanted variations in pixel values that do not correspond to actual trees. To address this, we apply Gaussian blur, a smoothing technique that reduces noise by averaging pixel values with their neighbors. This step ensures that the NDVI map is cleaner and more representative of the underlying vegetation patterns.

Once the NDVI map is smoothed, the next step is to convert it into a binary mask that clearly separates trees from non-trees. This is achieved through a method called Otsu's Thresholding, which automatically determines the optimal threshold value to divide the image into two categories: vegetation (trees) and non-vegetation. The output of this process is a black-and-white image where white pixels represent tree-covered areas, and black pixels represent all other features. This binary mask serves as a foundational output, enabling further analysis and visualization of urban greenery.

In summary, NDVI provides a powerful and intuitive way to detect vegetation in urban images. By using the distinct light reflectance properties of trees, NDVI transforms raw image data into meaningful insights about green cover. The additional steps of blurring and thresholding refine the results, ensuring that the final tree mask is both accurate and reliable for applications in urban planning and environmental monitoring.

4.3 DETECTING BUILDINGS USING K-MEANS CLUSTERING

The detection of buildings within urban landscapes is a critical aspect of urban analysis, as buildings define the structural and functional aspects of cities. Unlike trees, buildings lack specific and consistent color characteristics that can be easily identified. Therefore, a more flexible and adaptive approach is required to detect them. For this purpose, K-Means Clustering, a machine learning algorithm, is employed. This algorithm is well-suited for tasks involving segmentation, where the goal is to group similar data points into distinct categories.

K-Means Clustering works by dividing the pixels of an image into a predefined number of groups, referred to as k. For the purpose of building detection, we set k=3, representing three distinct clusters that typically correspond to vegetation, buildings, and other features (such as roads or open areas). Each pixel in the image is assigned to one of these clusters based on its color intensity and similarity to other pixels. For instance, pixels that share similar grayish shades, often indicative of building materials, are grouped together into the same cluster. This clustering process allows us to separate buildings from other elements in the image in a data-driven manner, without relying on fixed color thresholds.

Once the clustering is complete, the cluster corresponding to buildings is identified and extracted. This extracted cluster is converted into a binary mask, where white pixels represent building areas, and black pixels represent non-building areas. However, the raw output from clustering may contain noise and imperfections, such as small gaps or scattered false positives, which need to be addressed to improve accuracy.

To refine the building mask, we apply morphological operations, which are image processing techniques designed to clean and enhance binary images. Two key operations are used: closing and opening. Closing involves filling small gaps or holes within building regions, ensuring that the mask accurately represents complete structures. Conversely, opening removes small noise elements, such as isolated pixels that do not belong to buildings. By combining these operations, we produce a cleaner and more accurate building mask that faithfully represents the actual building regions in the urban image.

In summary, K-Means Clustering provides a robust and adaptable approach to building detection by grouping pixels based on their similarities, without relying on predefined color characteristics. The subsequent use of morphological operations ensures that the final building mask is both accurate and noise-free, making it suitable for applications in urban planning, infrastructure analysis, and land-use management. This process highlights the power of combining machine learning algorithms with image processing techniques for effective urban feature detection.

4.4 COMBINING TREE AND BUILDING MASKS

The process of combining tree and building masks into a single mask is an essential step in urban landscape analysis. While individual masks for trees and buildings are useful on their own, a combined mask offers a more comprehensive view of the urban environment. This combined representation is particularly valuable for real-world applications, such as urban planning, where decision-makers need to understand the spatial relationship between green spaces and built-up areas. By visualizing these features together, planners can identify areas where development can occur while ensuring that greenery is preserved or enhanced, ultimately promoting sustainable urban development.

Creating a combined mask involves assigning a unique label to each feature, making them visually and computationally distinct. For example, tree regions are labeled with the value 1, while building regions are labeled with the value 2. This labeling ensures that each feature type can be identified independently, even when they overlap or are in close proximity. The process involves overlaying the binary masks for trees and buildings, with the combined mask containing these unique labels for each pixel based on its classification.

This combined mask is then presented as a single image, where the labeled features are displayed in different colors or intensities to make them visually distinct. For example, trees might be displayed in green, buildings in gray, and other areas left uncolored or black. This visual distinction allows users to easily interpret the spatial distribution of green spaces and built-up areas, facilitating tasks like identifying areas for urban expansion, evaluating the balance between natural and developed spaces, or designing green corridors within the city.

By integrating tree and building masks into a single representation, this step bridges the gap between individual feature detection and holistic urban analysis. It provides a clear, actionable overview of the urban landscape, enabling data-driven decisions that balance development with environmental sustainability. This approach highlights the importance of combining segmentation outputs into a unified framework for effective urban management and planning.

4.5 SEGMENTATION

Segmentation is a critical step in the project, focusing on dividing urban imagery into meaningful components such as trees, buildings, and other urban features. This step involves processing high-resolution satellite images to assign each pixel a label corresponding to its category. The segmentation models used include U-Net and SegNet, each contributing unique strengths to handle the complexities of urban landscapes.

Algorithm 4.2 U-net Algorithm

Require: Input image I of shape (H,W,C), Number of classes *num_classes*
Ensure: Segmentation map S of shape $(H,W,num\_classes)$

1: Create the input layer *inputs* with shape (H,W,C)
▷ Encoder
2: **for** each layer in encoder blocks (4 layers) **do**
3: Apply two Conv2D layers with ReLU activation
4: Apply MaxPooling2D for downsampling
5: Store intermediate outputs for skip connections
6: **end for**
▷ Bottleneck
7: Apply two Conv2D layers with ReLU activation at the bottleneck
▷ Decoder
8: **for** each layer in decoder blocks (4 layers) **do**
9: Apply UpSampling2D for upsampling
10: Concatenate the upsampled output with the corresponding encoder output
11: Apply two Conv2D layers with ReLU activation
12: **end for**
▷ Output Layer
13: Apply a Conv2D layer with softmax activation to output a segmentation map S of shape $(H,W,num\_classes)$
14: **return** The segmentation map S

The U-Net model as seen in Algorithm 4.2 employs an encoder-decoder architecture with skip connections, enabling it to capture fine-grained spatial details while preserving contextual information. It processes the input image through a series of convolutional and max-pooling layers in the encoder, compressing the spatial dimensions and extracting hierarchical features. The decoder then reconstructs the segmentation map using upsampling layers and concatenated skip connections from the encoder, ensuring precise localization of features like trees and buildings.

SegNet as seen in Algorithm 4.3, designed for efficiency, uses a symmetric encoder-decoder architecture with pooling indices to minimize memory requirements and computational costs. It achieves lightweight segmentation without significant accuracy loss, making it ideal for real-time urban analysis. The encoder extracts spatial features, while the decoder reconstructs the segmentation map using the stored pooling indices, ensuring effective pixel-wise classification.

Algorithm 4.3 SegNet Algorithm

Require: Input image I of shape (H, W, C), Number of classes *num_classes*
Ensure: Segmentation map S of shape $(H, W, num\_classes)$

Create the input layer *inputs* with shape (H, W, C)
▷ Encoder (Downsampling Path)
for each layer in encoder blocks (4 blocks) **do**
    Apply Conv2D with appropriate filters (64, 128, 256, 512) and 3x3 kernel
    Apply BatchNormalization
    Apply ReLU activation
    Apply MaxPooling2D with 2x2 pool size and strides
    Store max-pooling indices for upsampling
end for
▷ Decoder (Upsampling Path)
for each layer in decoder blocks (4 blocks) **do**
    Apply UpSampling2D with 2x2 size using stored max-pooling indices
    Apply Conv2D with appropriate filters (512, 256, 128, 64) and 3x3 kernel
    Apply BatchNormalization
    Apply ReLU activation
end for
▷ Output Layer
Apply Conv2D (1x1 kernel, *num_classes* filters, softmax activation) to produce segmentation map S
return The segmentation map S of shape $(H, W, num\_classes)$

Multispectral satellite imagery combining RGB and Near-Infrared (NIR) bands is used to improve segmentation performance. RGB bands capture visible features like buildings, while the NIR band enhances vegetation detection through spectral reflectance properties. The models are trained using the Adam optimizer to achieve convergence and robustness in segmentation performance.

4.6 CLASSIFICATION

Classification is performed after segmentation to accurately label each segmented region into categories such as trees, buildings, and other urban features. This step translates the pixel-level segmentation results into meaningful feature categories for applications like urban planning and environmental monitoring.

The tree classification process uses the Normalized Difference Vegetation Index (NDVI), a widely used metric in remote sensing. NDVI is calculated using the green and red spectral bands of the imagery, leveraging the unique reflectance properties of vegetation. High NDVI values correspond to vegetation, enabling precise identification of tree-covered areas. Binary masks generated through NDVI are refined using Gaussian blurring to reduce noise and thresholding techniques to produce clear, interpretable outputs.

For building classification, K-Means Clustering is applied to segment image pixels based on intensity and similarity. By clustering pixels into predefined groups (e.g., vegetation, buildings, and other), the algorithm identifies regions corresponding to buildings. These regions are further refined using morphological operations like closing (to fill gaps) and opening (to remove noise), ensuring clean and accurate building masks.

Once trees and buildings are classified, the results are combined into a unified map to provide a holistic view of urban landscapes. This combined map assigns unique labels or colors to each feature type, allowing for easy interpretation and analysis.

4.7 VISUALIZING THE RESULTS

The process of visualizing the results is an essential part of urban landscape analysis, as it transforms technical outputs like binary masks into a format that is easy to interpret and understand. Binary masks, while effective for machine processing, can appear abstract and difficult to interpret for human users, particularly for stakeholders like urban planners or policymakers who may not have a technical background. Visualization bridges this gap by converting the masks into meaningful and intuitive visual outputs.

Visualization typically involves overlaying the generated masks onto the original urban images or displaying the masks side-by-side with the original images. This approach provides context by showing how the detected features—such as trees and buildings—align with their real-world locations. By visualizing these results, it becomes easier to evaluate the accuracy and effectiveness of the segmentation process and to identify any areas that require refinement.

One effective method of visualization is using color-coded images. For instance, trees in the mask can be represented in green, while buildings are shown in blue or gray. This color-coding makes it immediately clear which regions correspond to natural features and which represent built-up areas. The colors not only make the results visually appealing but also enhance the interpretability of the data, allowing users to quickly assess the balance between green spaces and infrastructure in an urban environment.

This visualization is particularly valuable for non-technical stakeholders, as it simplifies complex data into an intuitive visual format. For example, city planners can use the visualized outputs to identify areas with insufficient greenery, assess the density of buildings, or plan the placement of future developments. By making the data accessible and easy to understand, visualization ensures that the results of the analysis can be effectively used in real-world decision-making.

In summary, visualizing the results transforms raw segmentation outputs into actionable insights. By overlaying masks, using color-coded images, or presenting side-by-side comparisons, visualization ensures that the outputs are both accurate and interpretable. This step plays a critical role in making urban analysis inclusive, allowing technical and non-technical users alike to understand and act on the data effectively.

4.8 EVALUATING THE RESULTS

The evaluation of results is a critical stage in any analysis pipeline, especially for segmentation tasks in urban landscape analysis. This step involves assessing the accuracy of the generated masks for trees and buildings by comparing them to the ground truth, which consists of manually labeled masks. Ground truth data serves as a benchmark, providing the ”correct” segmentation against which the model’s predictions are validated. This evaluation is essential to determine how well the process has performed and to identify areas for improvement.

The primary goal of evaluation is to ensure that the segmentation models are functioning as intended. If the generated masks deviate significantly from the ground truth, it indicates that the model or preprocessing steps may require refinement. For instance, noisy results in the tree masks might suggest that the NDVI calculation or thresholding parameters need adjustment.

Similarly, inaccuracies in building detection could point to the need for tuning the clustering algorithm or morphological operations. Evaluation provides actionable insights, enabling iterative improvements to enhance the reliability and accuracy of the process.

Several metrics are used to quantify the performance of the segmentation models, each offering unique insights. One such metric is Intersection over Union (IoU), which measures the overlap between the predicted mask and the ground truth. IoU is calculated as:

$$\text{IoU} = \frac{\text{Intersection of Prediction and Ground Truth}}{\text{Union of Prediction and Ground Truth}} \tag{4.2}$$

A higher IoU indicates a better match between the predicted and actual regions, highlighting the model's spatial accuracy.

Another important metric is Precision, which evaluates how many of the predicted positives (e.g., detected trees or buildings) are correct. Precision is calculated as:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{4.3}$$

High precision indicates that the model minimizes false positives, which is crucial for accurate feature detection in urban environments.

The F1-Score combines precision and recall into a single metric, providing a balanced measure of the model's performance, especially when the dataset has an uneven distribution of features. It is calculated as:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.4}$$

This metric is particularly useful when both false positives and false negatives need to be considered equally important.

Lastly, the Dice Coefficient, which is similar to IoU, measures the similarity between the predicted mask and the ground truth. It is defined as:

$$\text{Dice} = \frac{2 \times \text{Intersection of Prediction and Ground Truth}}{\text{Total Pixels in Prediction + Total Pixels in Ground Truth}} \tag{4.5}$$

The Dice Coefficient is especially valuable for assessing the accuracy of segmenting objects with small or intricate shapes, such as individual trees or narrow building edges.

In summary, evaluation provides a quantitative understanding of how well the segmentation models perform. Metrics like IoU, Precision, F1-Score, and Dice Coefficient offer a comprehensive assessment of spatial accuracy, classification reliability, and overall model performance. By identifying strengths and weaknesses through evaluation, the process can be iteratively refined to ensure that the results meet the requirements for real-world applications like urban planning and environmental monitoring.

4.9 SOFTWARE REQUIREMENTS

This project involves multiple components for urban feature segmentation, classification, and map updating. The implementation includes various stages such as preprocessing satellite imagery, training and evaluating deep learning models, and visualizing outputs for urban analysis. This section elaborates upon various software requirements specific to different phases of the project.

4.9.1 Programming Language

Python: The primary programming language used for implementing the deep learning models, segmentation algorithms, and classification workflows. Python is chosen for its extensive ecosystem of deep learning frameworks like TensorFlow and PyTorch, as well as its capabilities in handling large-scale data processing and geospatial analysis, making it highly suitable for this project.

4.9.2 Required Libraries

NumPy, a fundamental library for numerical computing, used for handling arrays, performing mathematical operations, and structuring image data for deep learning models.

OpenCV, an open-source computer vision library, used for image processing tasks such as reading, resizing, thresholding, and applying pixel-level transformations, essential for segmentation and classification workflows.

Matplotlib, a visualization library used for plotting and displaying input images, segmentation masks, and model outputs, facilitating the interpretation and debugging of results.

TensorFlow and Keras, a deep learning framework employed for building and training models, including U-Net and DeepLabV3+, used for urban feature segmentation and classification.

Scikit-learn, a machine learning library used for data splitting, evaluation, clustering, and classification tasks. It facilitates tasks such as dataset partitioning, performance analysis, and clustering image pixels for segmentation workflows.

4.9.3 Development Tool

IDE or Editor used here is kaggle, an online platform for data science and machine learning, is used as the primary development environment. It provides a cloud-based interface with pre-installed libraries and GPUs, enabling seamless implementation and testing of deep learning models. Kaggle's notebook features support collaborative development and efficient workflow management.

CHAPTER 5

RESULTS AND DISCUSSION

In this chapter, the tools used and the results of the implemented modules, such as urban feature segmentation and classification for map updates, are presented. Furthermore, the advantages and limitations of the approach are discussed in detail.

5.1 MODEL EVALUATION RESULTS

The results of the U-Net and Seg-Net models are summarized, highlighting their performance on test datasets. Metrics include test accuracy, test loss, precision, recall, F1 score, and Intersection over Union (IoU).

Table 5.1: Overall Performance Metrics of U-Net and Seg-Net Models

Metric	U-Net	Seg-Net
Test Accuracy	65.32%	63.52%
Test Loss	0.7439	0.8014
Precision (Macro Avg)	0.7321	0.5755
Recall (Macro Avg)	0.6532	0.6352
F1 Score (Macro Avg)	0.5581	0.5451
IoU (Intersection over Union)	0.4449	0.4273

The results in Table 5.1 highlight that U-Net consistently outperformed Seg-Net across all metrics, demonstrating its superior performance in urban feature segmentation tasks.

U-Net achieved a higher test accuracy of 65.32%, surpassing Seg-Net's 63.52%. Additionally, U-Net recorded a lower test loss of 0.7439, compared to 0.8014 for Seg-Net, indicating its better optimization during training. The precision, recall, and F1 score metrics further emphasize U-Net's ability to classify and segment urban features accurately across multiple classes.

Moreover, the Intersection over Union (IoU), which evaluates the overlap between predicted and ground truth masks, was higher for U-Net (0.4449) compared to Seg-Net (0.4273), showcasing its effectiveness in achieving accurate segmentation results.

These results underscore U-Net's robust architecture, making it more efficient in preserving spatial details and achieving consistent segmentation outcomes compared to Seg-Net.

Table 5.2: Detailed Classification Report for U-Net and Seg-Net Models

Class / Metric	**U-Net**	**Seg-Net**
Dice Coefficient		
Class 0	0.7765	0.7671
Class 1	0.5401	0.4302
Class 2	0.0000	0.0432
Precision		
Class 0	0.66	0.65
Class 1	0.63	0.57
Class 2	0.00	0.39
Recall		
Class 0	0.95	0.95
Class 1	0.47	0.34
Class 2	0.00	0.02
F1 Score		
Class 0	0.78	0.77
Class 1	0.54	0.43
Class 2	0.00	0.04

The detailed classification report in Table 5.2 provides class-specific performance metrics for U-Net and Seg-Net models, highlighting their abilities to segment urban features, including background, trees, and buildings. The Dice

Coefficient shows that both models performed well for Class 0 (Background), with U-Net scoring 0.7765 and Seg-Net scoring 0.7671, indicating their proficiency in identifying background pixels. For Class 1 (Trees), U-Net outperformed Seg-Net with a Dice Coefficient of 0.5401, compared to 0.4302, reflecting its superior ability to capture vegetative regions. However, both models struggled with Class 2 (Buildings), where U-Net scored 0.0000 and Seg-Net slightly better at 0.0432, revealing a significant area for improvement.

When evaluating Precision, both models performed similarly for Class 0 (Background), with scores of 0.66 for U-Net and 0.65 for Seg-Net. For Class 1 (Trees), U-Net achieved 0.63, surpassing Seg-Net's 0.57, demonstrating better accuracy in identifying tree-covered areas. Seg-Net performed marginally better in Class 2 (Buildings), with a precision of 0.39 compared to U-Net's 0.00, though the overall performance for this class remained low. Recall scores for both models were high for Class 0 (Background), reaching 0.95, while U-Net showed better recall for Class 1 (Trees) at 0.47, compared to Seg-Net's 0.34. Both models showed poor recall for Class 2 (Buildings), with U-Net scoring 0.00 and Seg-Net scoring 0.02.

The F1 Score, which balances precision and recall, mirrored these trends. U-Net slightly outperformed Seg-Net for Class 0 (Background) with a score of 0.78 versus 0.77, and significantly outperformed Seg-Net for Class 1 (Trees) with a score of 0.54 compared to 0.43. For Class 2 (Buildings), both models performed poorly, with U-Net scoring 0.00 and Seg-Net achieving 0.04. These results underline U-Net's superior performance for background and tree segmentation, while both models face challenges in building segmentation, likely due to class imbalance or insufficient feature representation. These findings emphasize the need for future improvements, such as data augmentation or advanced loss functions, to enhance segmentation accuracy for underrepresented classes.

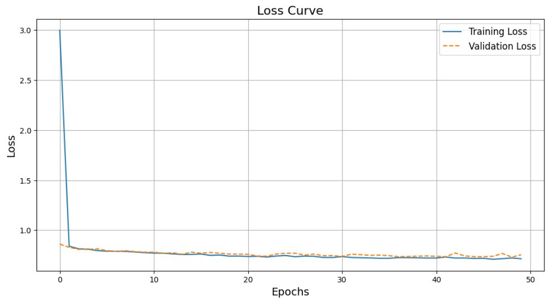

Figure 5.1: Training and Validation Loss Curves for U-Net.

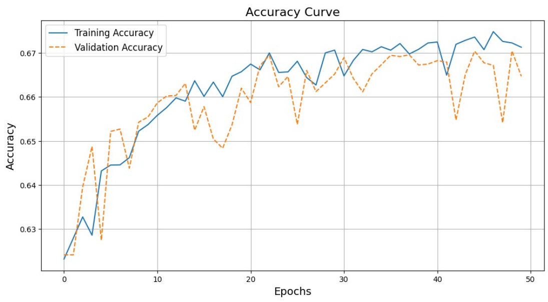

Figure 5.2: Training and Validation Accuracy Curves for U-Net.

The training and validation loss curves demonstrate the model's effective learning behavior, with a steep decline during initial epochs and stabilization in later stages. The training loss consistently remains lower than the validation loss, as shown in 5.1, indicating a good fit to the data and strong generalization without significant overfitting.

The accuracy curves reveal steady improvement in correctly segmenting urban features like trees and buildings, stabilizing around 67% in later epochs. As depicted in 5.2, the close alignment of training and validation accuracy curves highlights the model's robustness and ability to generalize effectively, despite minor fluctuations in validation accuracy.

Overall, the U-Net model proves well-suited for urban feature segmentation tasks, with stable and converging metrics reflecting its reliability and effectiveness in addressing segmentation challenges.

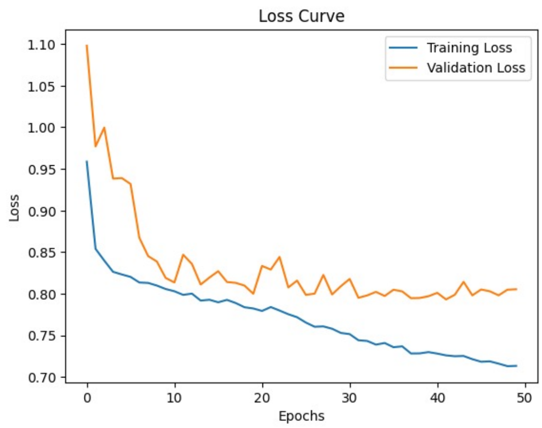

Figure 5.3: Training and Validation Loss Curves for Seg-Net.

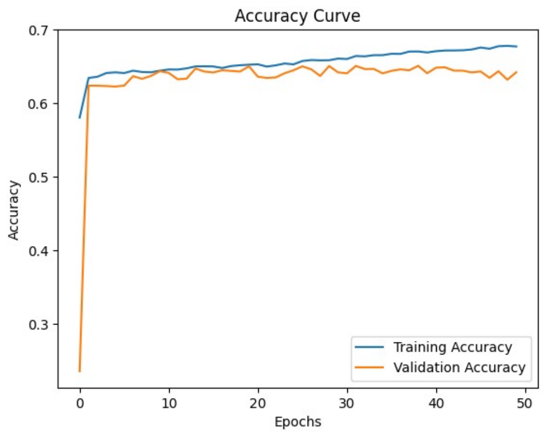

Figure 5.4: Training and Validation Accuracy Curves for Seg-Net.

The Seg-Net model's performance during training and validation is shown through loss and accuracy curves over 50 epochs. The loss curve highlights a significant reduction in both training and validation loss during initial epochs, with training loss stabilizing around 0.70 and validation loss near 0.80. The minimal gap between the two curves, as shown in 5.3, indicates effective learning and good generalization without significant overfitting.

The accuracy curve, depicted in 5.3, shows steady improvement, with training accuracy reaching 70% and validation accuracy stabilizing near 65%. This close alignment between training and validation accuracy reflects the model's robustness and ability to generalize effectively across datasets, despite minor fluctuations.

Overall, the Seg-Net model demonstrates strong learning and generalization capabilities for urban feature segmentation tasks. These results highlight its potential for practical applications like urban map updates and environmental monitoring. Further enhancements, such as fine-tuning hyperparameters and augmenting data, could improve performance and reduce validation variability.

5.2 COMPARATIVE ANALYSIS

The comparative analysis reveals U-Net as a more effective model for urban feature segmentation compared to Seg-Net. U-Net consistently achieved higher metrics, including accuracy, precision, and Intersection over Union (IoU), making it more suitable for real-world applications like urban planning and environmental monitoring. Its encoder-decoder architecture with skip connections allows for better retention of spatial details, resulting in improved segmentation performance across various classes.

However, both models faced challenges in accurately segmenting complex features, particularly buildings, due to overlapping textures and class imbalances. These limitations highlight the need for enhancements in data preparation, loss function design, and model architecture. Future improvements, such as incorporating diverse datasets, optimizing loss functions, and leveraging advanced techniques like attention mechanisms, can further improve segmentation performance and address current challenges effectively.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

This project successfully showcased the use of deep learning models, specifically U-Net and SegNet, for the segmentation and classification of critical urban features, such as trees and buildings, from multispectral satellite imagery. These models, with their robust architectures, demonstrated their ability to process complex urban environments and deliver pixel-level segmentation with high spatial precision. By employing advanced evaluation metrics like IoU, Precision, F1-Score, and Dice Coefficient, the models' performance was systematically analyzed, revealing their strengths in identifying and classifying features even in challenging scenarios with overlapping objects and diverse textures. The automated segmentation outputs generated through this pipeline present a reliable solution for updating urban maps efficiently, paving the way for practical applications in urban planning, environmental monitoring, and sustainable development projects.

Moreover, the use of multispectral imagery, combining RGB and NIR bands, enhanced the detection of vegetation and improved overall segmentation quality. This approach effectively addressed the limitations of traditional segmentation methods by leveraging the strengths of modern deep learning architectures. The results validate the applicability of the proposed methodology in real-world scenarios, highlighting its potential to transform urban feature detection and provide valuable data-driven insights to support decision-making processes in urban development.

6.2 FUTURE WORK

While this project successfully focused on the segmentation of trees and buildings, future work will aim to expand the system's capability to identify and segment a broader range of urban features, including roads, water bodies, parks, and open spaces. Integrating these additional features into the segmentation pipeline will enable the creation of more comprehensive urban maps, supporting detailed analyses for infrastructure development, disaster management, and environmental conservation. The inclusion of these features will also provide a holistic understanding of urban environments, ensuring more informed and sustainable urban planning.

Efforts will also focus on further improving the accuracy and efficiency of the models. This may involve incorporating advanced deep learning architectures, such as transformer-based segmentation models or hybrid approaches, to address current limitations and achieve finer granularity in segmentation outputs. Optimization techniques, including hyperparameter tuning, data augmentation, and larger, more diverse training datasets, will be explored to enhance the models' generalization to different urban scenarios.

Real-world validations will form a critical aspect of future work, allowing the system to be tested across diverse geographies and conditions. Additionally, the integration of real-time data feeds will transform the static segmentation process into a dynamic system capable of adapting to changes in urban landscapes. By addressing these aspects, the system will evolve into a more robust and versatile tool for urban analysis, supporting more effective and efficient urban development strategies in the future.

REFERENCES

[1] Jonathan Ventura, Camille Pawlak, Milo Honsberger, Cameron Gonsalves, Julian Rice, Natalie L.R. Love, Skyler Han, Viet Nguyen, Keilana Sugano, Jacqueline Doremus, G. Andrew Fricker, Jenn Yost, and Matt Ritter. Individual tree detection in large-scale urban environments using high-resolution multispectral imagery. *International Journal of Applied Earth Observation and Geoinformation*, 130:103848, 2024.

[2] Matheus Pinheiro Ferreira, Daniel Rodrigues dos Santos, Felipe Ferrari, Luiz Carlos Teixeira Coelho Filho, Gabriela Barbosa Martins, and Raul Queiroz Feitosa. Improving urban tree species classification by deep-learning based fusion of digital aerial images and lidar. *Urban Forestry Urban Greening*, 94:128240, 2024.

[3] Burcu Amirgan and Arzu Erener. Semantic segmentation of satellite images with different building types using deep learning methods. *Remote Sensing Applications: Society and Environment*, 34:101176, 2024.

[4] Abdoulie Fatty, An-Jui Li, and Chih-Yuan Yao. Instance segmentation based building extraction in a dense urban area using multispectral aerial imagery data. *Multimedia Tools and Applications*, 83(22):61913–61928, 2024.

[5] Sultan Daud Khan, Louai Alarabi, and Saleh Basalamah. An encoder–decoder deep learning framework for building footprints extraction from aerial imagery. *Arabian Journal for Science and Engineering*, 48(2):1273–1284, 2023.

[6] Xihong Lian, Limin Jiao, Zejin Liu, Qiqi Jia, Wei Liu, and Yaolin Liu. A detection of street trees and green space: Understanding contribution of urban trees to climate change mitigation. *Urban Forestry Urban Greening*, 102:128561, 2024.

[7] Rami Al-Ruzouq, Mohamed Barakat A. Gibril, Abdallah Shanableh, Jan Bolcek, Fouad Lamghari, Nezar Atalla Hammour, Ali El-Keblawy, and Ratiranjan Jena. Spectral–spatial transformer-based semantic segmentation for large-scale mapping of individual date palm trees using very high-resolution satellite data. *Ecological Indicators*, 163:112110, 2024.

[8] Anqi Lin, Xiaomeng Sun, Hao Wu, Wenting Luo, Danyang Wang, Dantong Zhong, Zhongming Wang, Lanting Zhao, and Jiang Zhu. Identifying urban building function by integrating remote sensing imagery and poi data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:8864–8875, 2021.

[9] Zoe Mayer, James Kahn, Yu Hou, Markus Götz, Rebekka Volk, and Frank Schultmann. Deep learning approaches to building rooftop thermal bridge detection from aerial images. *Automation in Construction*, 146:104690, 2023.

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

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

[12] Rao Li, GuoDong Sun, Sheng Wang, TianZhuzi Tan, and Fu Xu. Tree trunk detection in urban scenes using a multiscale attention-based deep learning method. *Ecological Informatics*, 77:102215, 2023.

[13] Abubakar Sani-Mohammed, Wei Yao, and Marco Heurich. Instance segmentation of standing dead trees in dense forest from aerial imagery using deep learning. *ISPRS Open Journal of Photogrammetry and Remote Sensing*, 6:100024, 2022.

[14] Chun Liu, Doudou Zeng, Akram Akbar, Hangbin Wu, Shoujun Jia, Zeran Xu, and Han Yue. Context-aware network for semantic segmentation toward large-scale point clouds in urban environments. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–15, 2022.

[15] Chiara Romanengo, Daniela Cabiddu, Simone Pittaluga, and Michela Mortara. Building semantic segmentation from large-scale point clouds via primitive recognition. *Graphical Models*, 136:101234, 2024.

[16] Saqib Ali Khan, Yilei Shi, Muhammad Shahzad, and Xiao Xiang Zhu. Fgcn: Deep feature-based graph convolutional network for semantic segmentation of urban 3d point clouds. pages 778–787, 2020.

[17] R. Shanmuga Priya and K. Vani. Climate change forecast for forest fire risk prediction using deep learning. In *2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS)*, volume 1, pages 1065–1070, 2023.

[18] R. Shanmuga Priya and K. Vani. Burn severity estimation in mendocino national forest using bma model and sentinel images. In *2024 International Conference on Advances in Computing, Communication and Applied Informatics (ACCAI)*, pages 1–6, 2024.