DAYANANDA SAGAR UNIVERSITY

Devarakaggalahalli, Harohalli, Kanakapura Road, Ramanagara Dt, Bengaluru-562112, Karnataka, India

CAPSTONE PROJECT PHASE - II REPORT ON

"Urban Growth Forecasting and LULC Dynamics in

Bangalore using Random Forest Classification and

Cellular Automata on Dynamic World Satellite Data"

Submitted in Partial fulfillment for award of degree in

Bachelor of Technology
in
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE & MACHINE LEARNING)

Submitted by

SHRUTI NIGAM
(ENG21AM012)
RISHITHA
KATTIPALLEM
(ENG21AM095)

Under The Supervision of

Prof. Bhuvana Mohini TN
Assistant Professor
Dept. of CSE(AI&ML),
School of Engineering
Dayananda Sagar University

Dr. Rangaraj BS
Research Professor
Dept. of CSE(AI&ML)
School of Engineering
Dayananda Sagar University

DAYANANDA SAGAR UNIVERSITY

Department of Computer Science & Engineering
(Artificial Intelligence & Machine Learning)
Devarakaggalahalli, Harohalli, Kanakapura Road, Ramanagara Dt,
Bengaluru-562112, Karnataka, India

CERTIFICATE

This is to certify that the Capstone Project - II report, entitled **"Urban Growth Forecasting and LULC Dynamics in Bangalore using Random Forest Classification and Cellular Automata on Dynamic World Satellite Data"**, completed by **Shruti Nigam (ENG21AM0120), Rishitha Kattipallem (ENG21AM0095),** bonafide students of Bachelor of Technology in Computer Science and Engineering (Artificial Intelligence & Machine Learning) at the School of Engineering, Dayananda Sagar University, Bengaluru in partial fulfillment of the requirement for the VIII semester during the academic year 2024-2025.

Prof. Bhuvana Mohini	**Dr. Vinutha N**	**Chairperson**
Assistant Professor	Project Coordinator	Dr. Jayavrinda Vrindavanam
Dept. of CSE(AI&ML)	Associate Professor	Professor
School of Engineering	Dept. of CSE(AI&ML),	Dept. of CSE(AI&ML),
Dayananda Sagar University	School of Engineering	School of Engineering
	Dayananda Sagar University	Dayananda Sagar University
Date:	**Date:**	**Date:**

Name of Examiner | **Signature of Examiner with date**

1.

2.

DAYANANDA SAGAR UNIVERSITY

Department of Computer Science & Engineering
(Artificial Intelligence & Machine Learning)
Devarakaggalahalli, Harohalli, Kanakapura Road,
Ramanagara Dt, Bengaluru-562112, Karnataka, India

DECLARATION

We, **Shruti Nigam (ENG21AM0120), Rishitha Kattipallem (ENG21AM0095)**, students of the eighth semester of B.Tech in Computer Science and Engineering (AI & ML), at School of Engineering, Dayananda Sagar University, hereby declare that the Capstone Project Phase 2 titled **"Urban Growth Forecasting and LULC Dynamics in Bangalore using Random Forest Classification and Cellular Automata on Dynamic World Satellite Data"** has been carried out by us and submitted in partial fulfillment for the award of degree in Bachelor of Technology in Computer Science and Engineering (AI & ML) during the academic year 2024-2025.

Place : Bengaluru
Date :

ACKNOWLEDGEMENT

It is a great pleasure for us to acknowledge the assistance and support of many individuals who have been responsible for the successful completion of this project work.

First, we take this opportunity to express our sincere gratitude to School of Engineering, Dayananda Sagar University, Bengaluru, Karnataka, India for providing us with a great opportunity to pursue my Bachelor's degree in this institution.

We am highly elated in expressing my sincere and abundant respect to ***Dr. D. Hemachandra Sagar****, Chancellor,* ***Dr. D. Premachandra Sagar****, Pro Chancellor,* ***Dr. Amit R Bhatt****, Vice Chancellor,* ***Dr. S. Prakash****, Pro-Vice Chancellor,* ***Dr. C. Puttamadappa****, Registrar,* ***Dr. K. R. Udaya Kumar Reddy,*** *Dean, School of Engineering (SoE)****,*** *Dayananda Sagar University, Bengaluru, Karnataka, India for their constant encouragement and expert advice.*

It is a matter of immense pleasure to express oursincere thanks to ***Dr. Jayavrinda Vrindavanam, Professor & Department Chairperson****,* ***Computer Science and Engineering (Artificial Intelligence & Machine Learning)****,* ***Dayananda Sagar University, Bengaluru, Karnataka, India*** *for providing right academic guidance that made our task possible.*

It is a matter of immense pleasure to express our sincere thanks to our guides, ***Dr. Rangaraj BS, Research Professor, and Prof. Bhuvana Mohini, Assistant Professor, Computer Science, and Engineering (Artificial Intelligence and Machine Learning), School of Engineering, Dayananda Sagar University****, for providing the right academic guidance that made this task possible and for sparing their valuable time to extend help in every step of our Capstone project work, which paved the way for smooth progress and the fruitful culmination of the research.*

We are also grateful to my family and friends who provided us with every requirement throughout the course.

We would like to thank one and all who directly or indirectly helped us in the Capstone Project-II work.

Shruti Nigam (ENG21AM0120)

Rishitha Kattipallem (ENG21AM0095)

CONTENTS

ABSTRACT

Urban growth forecasting is vital for sustainable development in rapidly expanding cities. This project presents a two-stage hybrid framework for land use and land cover (LULC) classification and urban expansion forecasting in Bangalore using Random Forest (RF) and Cellular Automata (CA). High-resolution Dynamic World V1 satellite data from 2020 to 2025, derived from Sentinel-2 imagery, was used. The RF model was trained on per-pixel class probability bands to classify land into nine LULC categories, achieving 98.21% overall accuracy, a Kappa coefficient of 0.9752, and a mean IoU of 0.9004. Key classes like trees and built-up showed particularly high IoU scores. The second stage employed a CA-based simulation using a 3×3 neighborhood kernel to forecast 2026 urban expansion from 2025 built-up clusters. A projected 6.21% growth in built-up area was estimated. Tree-to-built-up transitions were also analyzed, predicting a loss of 41.8 sq.km of tree cover. Backcasting validated the CA model with IoU scores over 0.86. Implemented in Google Earth Engine with Python libraries, the system offers a scalable and interpretable tool for environmental monitoring. This work supports policymakers in visualizing short-term urban sprawl and contributes to data-driven planning aligned with Sustainable Development Goals (SDGs).

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

Urbanization is one of the most transformative phenomena of the 21st century, particularly in rapidly growing cities like Bangalore, India. The uncontrolled expansion of urban areas leads to significant impacts on land use and land cover (LULC), especially the conversion of vegetated or agricultural land into built-up zones. Monitoring and forecasting such urban dynamics is essential for sustainable development, informed policymaking, ecological preservation, and infrastructure planning. Conventional approaches often rely on static labeled LULC maps and low-frequency satellite imagery, which fail to capture the short-term variability and per-pixel uncertainty present in modern urban environments.

This project addresses these limitations by proposing a hybrid machine learning and simulation-based framework for urban growth forecasting and LULC classification. It leverages Google's Dynamic World V1 dataset, which provides near real-time, 10-meter resolution LULC predictions derived from Sentinel-2 imagery. Uniquely, this dataset offers per-pixel class probability bands for nine LULC categories—water, trees, grass, flooded vegetation, crops, shrub/scrub, built, bare, and snow/ice—alongside a "label" band representing the most likely class. These probabilities allow for a richer representation of uncertainty and land cover mixtures, offering significant advantages over traditional single-label inputs.

The proposed system operates in two main stages. In the first stage, a Random Forest (RF) classifier is trained on the probability bands extracted from Dynamic World images for early 2025. A total of 5000 randomly sampled points across Bangalore were used, with a 70:30 split for training and testing. The RF model, an ensemble learning method known for its robustness and interpretability, was selected for its ability to handle high-dimensional input and prevent overfitting. The classifier produced a high-accuracy pixel-wise LULC map for 2025, validated using multiple metrics including overall accuracy (98.21%), Kappa coefficient (0.9752), and mean Intersection over Union (IoU) across classes (0.9004).

In the second stage, a spatial Cellular Automata (CA) simulation is applied to predict 2026 urban expansion. Using a 3×3 neighborhood kernel, the model identifies non-built-up pixels adjacent to built-up zones as potential growth candidates. This rule-based method reflects the natural outward sprawl behavior of urban development. The projected built-up area for 2026 shows a 6.21% increase compared to 2025, with 41.8 sq.km of tree cover at risk of conversion. To validate the CA simulation, a backcasting approach is employed—predicting 2021 and 2024 built-up areas using 2020 and 2023 maps respectively—yielding IoU scores above 0.86.

This project was implemented in Google Earth Engine (GEE) using Python libraries such as geemap and Earth Engine API. The interactive maps, feature importance analysis, and tree-to-built-up transition visualizations make the system both interpretable and accessible. The methodology supports SDG goals related to sustainable cities, climate action, and ecosystem conservation. By combining machine learning with spatial simulation and using modern satellite data, the proposed framework offers an effective decision-support tool for urban planners and environmental policymakers.

2 PROBLEM DEFINITION

Urban growth forecasting and land use/land cover (LULC) classification are critical challenges in environmental planning, particularly in fast-expanding urban areas such as Bangalore. As the population rises and cities expand outward, previously vegetated or agricultural lands are often converted into built-up zones. This transformation leads to increased pressure on infrastructure, environmental degradation, and the loss of green cover, particularly tree-dominated landscapes. Policymakers and urban planners require precise, timely, and interpretable spatial data to make informed decisions about zoning, development, conservation, and infrastructure investment.

Traditional approaches to LULC mapping and forecasting have relied heavily on static datasets, coarse-resolution imagery, or manual labeling techniques. Often, these methods utilize low-frequency satellite images and assign a single label per pixel without capturing uncertainty. Moreover, while some models use sophisticated classification methods, they do not incorporate spatial dynamics or simulate how urban areas naturally evolve over time. This limits their ability to support actionable short-term forecasting and early warning systems for urban sprawl.

Further, while Cellular Automata (CA) and Markov models have been widely used for LULC forecasting, many rely on historical land cover maps and deterministic transition matrices, failing to integrate the rich spatial-temporal information available from new satellite products. Similarly, many machine learning models applied in this domain depend on post-classified LULC labels rather than leveraging more informative features such as class probabilities, which can enhance model confidence and accuracy.

To address these gaps, this project defines the problem as twofold:

To build a highly accurate and interpretable land use and land cover classification model for Bangalore in early 2025, using the latest available data from Google's Dynamic World V1 collection. Unlike traditional datasets, Dynamic World provides not just labels but also per-pixel probability distributions for nine land cover classes. This enables a more nuanced classification that captures uncertainty and mixed pixels, particularly at urban–forest or urban–agriculture boundaries.

To simulate short-term urban growth from 2025 to 2026, focusing on the spatial expansion of built-up areas. This involves predicting which non-built-up pixels are most likely to transition based on their spatial proximity to current urban zones using Cellular Automata (CA). Additionally, the system should quantify the extent of ecological change by identifying tree-to-built-up conversions and validate the simulation logic using backcasting from earlier years.

A successful solution to this problem should be interpretable, scalable, and suitable for integration with environmental monitoring systems. It should allow not just classification but also simulation and validation, making it useful for real-world urban planning decisions. The system must also be efficient to execute within cloud-based platforms like Google Earth Engine, enabling near-real-time applications in dynamic urban contexts.

In summary, the defined problem is to create a technically sound, spatially detailed, and policy-relevant tool for LULC classification and short-term urban forecasting that bridges the gap between modern satellite data, machine learning classification, and rule-based spatial simulation.

3 LITERATURE SURVEY

Land use and land cover (LULC) monitoring and forecasting are essential tasks for sustainable urban development, especially in regions facing rapid expansion. Several studies have focused on LULC classification using satellite imagery, with methods ranging from traditional remote sensing indices to modern machine learning approaches. A commonly used dataset is Landsat imagery, which, despite its long temporal span, has a coarse resolution of 30 meters and lacks per-pixel uncertainty information [1].

Machine learning models such as Support Vector Machines (SVM), Decision Trees, and Random Forests (RF) have demonstrated strong performance in LULC classification due to their non-parametric nature and ability to handle high-dimensional data [2]. Among them, RF is often preferred for its ensemble nature, robustness to overfitting, and built-in feature importance metrics [3]. However, many of these models are trained on static LULC labels and fail to incorporate the rich class probabilities now available from datasets such as Google's Dynamic World.

Dynamic World V1, introduced by Google in collaboration with the World Resources Institute, provides near real-time 10-meter resolution land cover maps with per-pixel probability distributions across nine LULC classes [4]. This probabilistic information is valuable in capturing spatial uncertainty and mixed land cover zones, which are especially prevalent in transitional urban areas. However, its usage in machine learning–driven LULC classification remains underexplored.

Urban expansion modeling has also been extensively studied through Cellular Automata (CA) and CA-Markov frameworks. These models simulate spatial growth patterns based on neighborhood rules and temporal transition probabilities [5]. While CA is effective in capturing spatial spread (e.g., city sprawl), it often lacks integration with recent satellite data and is typically driven by deterministic transition matrices or binary maps. Studies such as [6] used the CA-Markov model to forecast long-term urban growth in regions like Al-Hassa, Saudi Arabia, based on NDVI/NDWI and MLC classification. However, they do not utilize modern probability-based LULC products like Dynamic World or machine learning-based classifiers.

A few recent works have attempted to integrate machine learning and CA for urban modeling. For instance, [7] explored a hybrid approach using RF and CA for urban expansion but still relied on manually generated features or outdated labeled datasets. Our work differentiates itself by using real-time, probabilistic LULC input, and validating CA-based growth forecasts through backcasting (e.g., 2020→2021).

This literature gap demonstrates the need for a framework that combines modern satellite products, machine learning classification, and spatial simulation, validated using real-world comparisons. Our work aims to fill this gap by integrating Dynamic World probability bands with Random Forest for classification, followed by a rule-based CA model to simulate urban growth.

4 PROJECT DESCRIPTION

This project presents a hybrid framework for urban growth forecasting and land use/land cover (LULC) classification in Bangalore, combining machine learning with spatial simulation. The methodology integrates Google's Dynamic World V1 satellite dataset with Random Forest classification and Cellular Automata (CA) simulation to predict short-term urban expansion and ecological transitions.

The primary dataset used is Dynamic World V1, which provides global 10-meter resolution LULC maps derived from Sentinel-2 imagery. Unlike traditional datasets that assign one fixed label per pixel, Dynamic World offers per-pixel class probabilities for nine classes: water, trees, grass, flooded vegetation, crops, shrub/scrub, built, bare, and snow/ice. This probability-rich data allows for more nuanced classification by capturing class uncertainty and mixed land cover characteristics, especially along urban-vegetation boundaries.

The first phase of the project involves LULC classification using a Random Forest (RF) model. A total of 5000 random sample points were generated within Bangalore's region of interest and split into 70% training and 30% testing sets. The input features to the classifier are the nine class probability bands, and the target label is the most likely class (label band). The Random Forest classifier, configured with 50 decision trees, was trained and evaluated using several metrics: overall accuracy (98.21%), Kappa coefficient (0.9752), and mean Intersection over Union (IoU) across classes (0.9004). The confusion matrix and feature importance plots further confirmed the robustness and interpretability of the model.

In the second phase, the focus shifts to forecasting built-up expansion for 2026 using a rule-based CA model. The binary built-up mask from the 2025 LULC classification serves as the base layer. A 3×3 circular kernel is applied to identify adjacent non-built-up pixels, which are marked as potential urban growth zones using focal spatial logic. The projected built-up area for 2026 is calculated by combining the original built-up mask and these growth candidates. This stage simulates how cities naturally sprawl outward from existing urban cores.

To validate the CA simulation, a backcasting approach was adopted—using 2020 to simulate 2021 and 2023 to simulate 2024. The forecasted maps were then compared with actual Dynamic World maps, and Intersection over Union (IoU) was calculated to assess spatial accuracy, achieving values above 0.86. Furthermore, by overlaying the forecasted 2026 built-up map with the 2025 tree mask, the system quantified tree-to-built-up transitions, predicting a 41.8 sq.km loss in tree cover. The forecast also showed a 6.21% increase in total built-up area from 2025 to 2026.

All components were implemented using Google Earth Engine (GEE) and Python-based libraries such as geemap, Earth Engine API, NumPy, and matplotlib. The results were visualized through interactive maps, RGB renderings, and heatmaps. The model supports environmental planning and SDG-aligned policy making by offering interpretable, accurate, and real-time forecasting of land dynamics. It also lays the groundwork for more advanced urban modeling by combining explainable AI with satellite-driven simulation.

5 METHODOLOGY

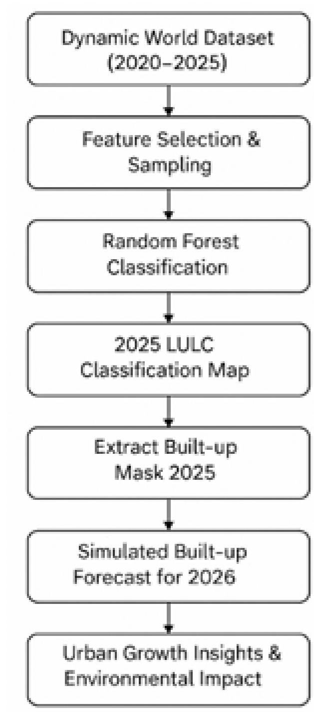


Fig. 1. System Architecture of the Proposed Model

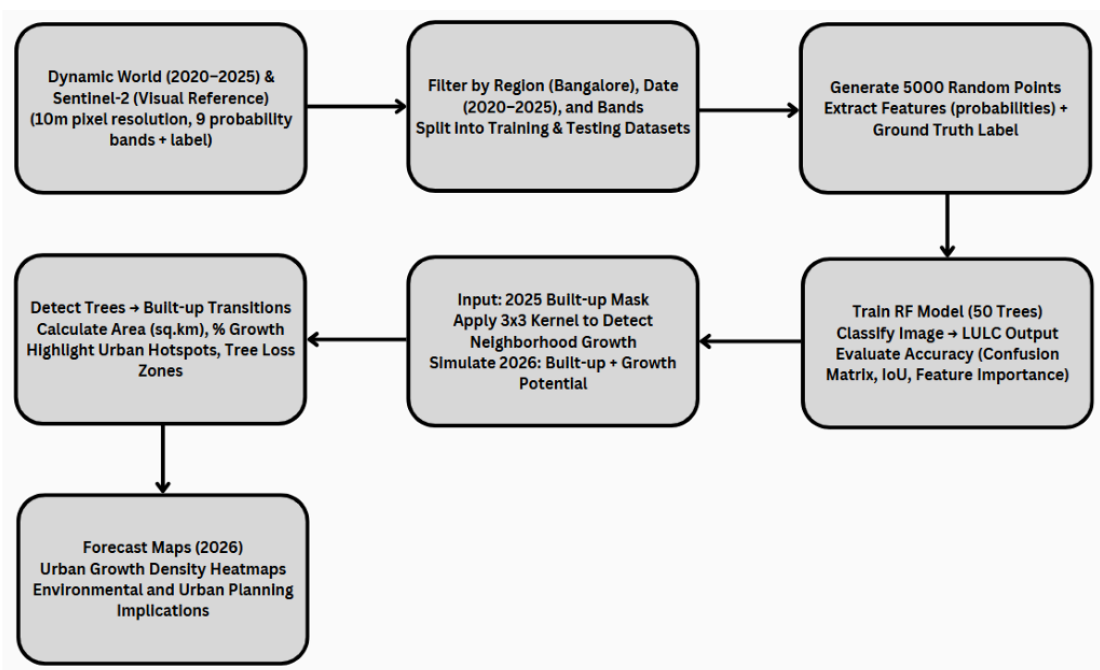


Fig. 2. Proposed Methodology for Urban Land Use Classification and Forecasting using Random Forest and Cellular Automata

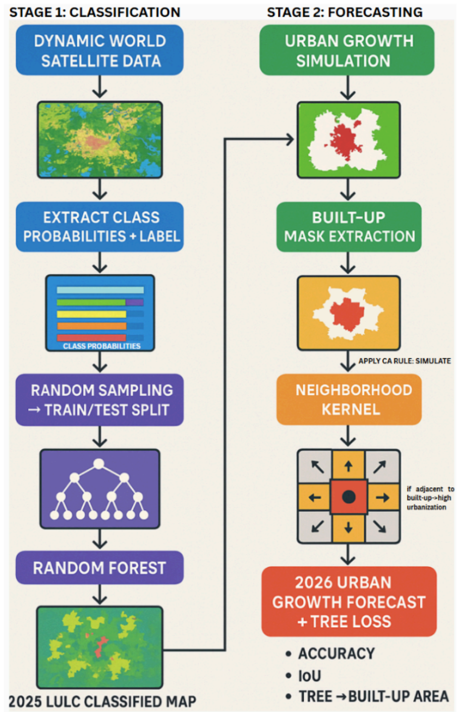


Fig. 3. Workflow of Framework

The goal of this study is to develop an intelligent and interpretable land use and land cover (LULC) classification and forecasting system for the Bangalore region. The system aims to identify different land cover types using high-resolution satellite data and accurately forecast urban expansion patterns—specifically built-up area growth—by 2026.

To achieve this, the system integrates: A supervised machine learning model (Random Forest) for pixel-wise classification of Dynamic World Sentinel-2 data, A Cellular Automata (CA)-based neighborhood expansion approach for simulating future urban growth based on spatial patterns observed from 2020 to 2025.

The solution should ensure: Effective classification of land cover categories (e.g., trees, water, built-up, etc.) using multi-band input features, Forecasting of built-up area growth using spatial transition rules that mimic real-world urban sprawl tendencies, Quantitative assessment of model performance through accuracy, confusion matrix, Kappa score, Intersection over Union (IoU), and backcasting validations, Environmental insight by quantifying tree-to-built-up conversions and spatial density of predicted growth.

The system uses the Google Earth Engine (GEE) cloud platform and Dynamic World V1 dataset for near real-time land cover classification. The primary objective is to build a lightweight, explainable, and scalable forecasting model suitable for aiding city planners and researchers concerned with sustainable urban expansion.

The two-stage process begins with LULC (Land Use Land Cover) classification, where Sentinel-2 based Dynamic World images (2020–2025) for the Bangalore region are used as input. Probability bands for all land cover classes and the top-1 label band are extracted as features. Labeled training and testing points are generated through random sampling, and a Random Forest classifier is trained to classify each pixel into one of the nine Dynamic World land cover classes. Model performance is evaluated using metrics such as accuracy, Kappa coefficient, confusion matrix, and IoU, resulting in a classified LULC map for early 2025. In the second stage, urban growth forecasting is performed using a built-up mask derived from the 2020–2025 LULC classification. A Cellular Automata (CA)-based neighborhood expansion rule is applied, where pixels adjacent to existing built-up areas are marked as potential growth zones. This leads to the generation of a projected built-up map for 2026, followed by visualization and quantification of changes such as built-up area increase, tree-to-built conversions, and growth density heatmaps.

Datasets

To model and forecast urban expansion using machine learning and spatial dynamics, this study relies on high-resolution, temporally consistent datasets from Google Earth Engine (GEE), specifically the Dynamic World (DW) and Sentinel-2 Harmonized datasets. The DW dataset, developed by Google and the World Resources Institute, provides near real-time 10-meter resolution land use/land cover (LULC) classifications with probabilistic estimates across nine major categories (e.g., trees, built-up, crops, water). Its “label” band, which identifies the most likely class, serves as a key input for supervised classification and temporal analysis. The dataset is central to our workflow—used for feature extraction, Random Forest (RF) training, and change detection.

Complementing this, Sentinel-2 Harmonized imagery contributes spectral bands for visual validation, cloud masking, and additional quality checks. Though DW is derived from Sentinel-2, direct access to raw bands enriches classifier training and visualization accuracy. Data is filtered to the Bangalore region from 2020 to 2025, and random sampling via GEE’s sample() function generates labeled points. These are split 70:30 into training and testing sets for performance evaluation using accuracy metrics and confusion matrices.

Despite its strengths, DW’s per-pixel classification may introduce noise in heterogeneous zones, and its use of maximum probability labels can oversimplify mixed-use areas. However, the dataset’s temporal granularity and spatial consistency make it highly suitable for LULC modeling. Together, DW and Sentinel-2 form a robust data foundation for simulating urban growth and assessing land transition dynamics over time.

6 EXPERIMENTATION

This study combines machine learning classification with Cellular Automata (CA)-based forecasting to model future land use and land cover (LULC) changes, focusing on urban expansion in Bangalore. The entire pipeline is implemented on Google Earth Engine (GEE) using Python and Geemap, ensuring spatial precision and scalability for urban sustainability research.

We utilize the Dynamic World (DW) dataset, which offers 10-meter resolution land cover data from January 2020 to April 2025, containing probabilistic estimates for nine LULC classes along with a "label" band indicating the most likely class. Sentinel-2 Harmonized imagery supports visual inspection and data quality checks. Data is spatially filtered to the Bangalore region, and relevant bands are selected.

Feature vectors are generated using the DW class probabilities, with the "label" band as ground truth. A random sampling approach is used to extract 5000 points, which are split into 70% training and 30% testing subsets. A Random Forest (RF) classifier with 50 trees is trained to predict LULC classes. Classification performance is evaluated using accuracy, confusion matrix, precision, recall, F1-score, and Intersection over Union (IoU). The model achieves over 98% accuracy, with high IoU values, particularly for built-up and trees. Feature importance analysis reveals that "trees," "crops," and "shrub_and_scrub" are most predictive.

Forecasting is performed using a simple yet effective CA model. A 3x3 kernel is used to identify potential growth zones adjacent to existing built-up areas using focal_max(). The 2026 map is generated by merging current built-up zones with these likely expansion pixels. This neighborhood-based logic simulates short-term urban growth where planning data is unavailable.

The complete pipeline—data acquisition, classification, and forecasting—is illustrated in Figure 2. The system also detects meaningful transitions such as "trees to built-up," offering valuable inputs for sustainability planning. Metrics like projected built-up growth and vegetation loss align with Sustainable Development Goals (SDGs), highlighting the model’s broader utility in urban policy and environmental monitoring.

7 RESULTS AND ANALYSIS

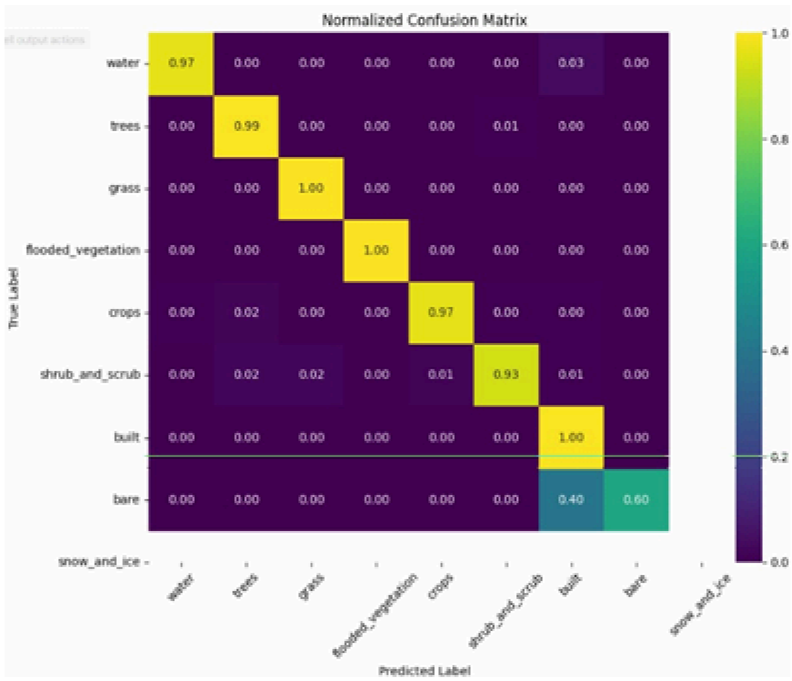


Fig. 3. Random Forest Confusion Matrix

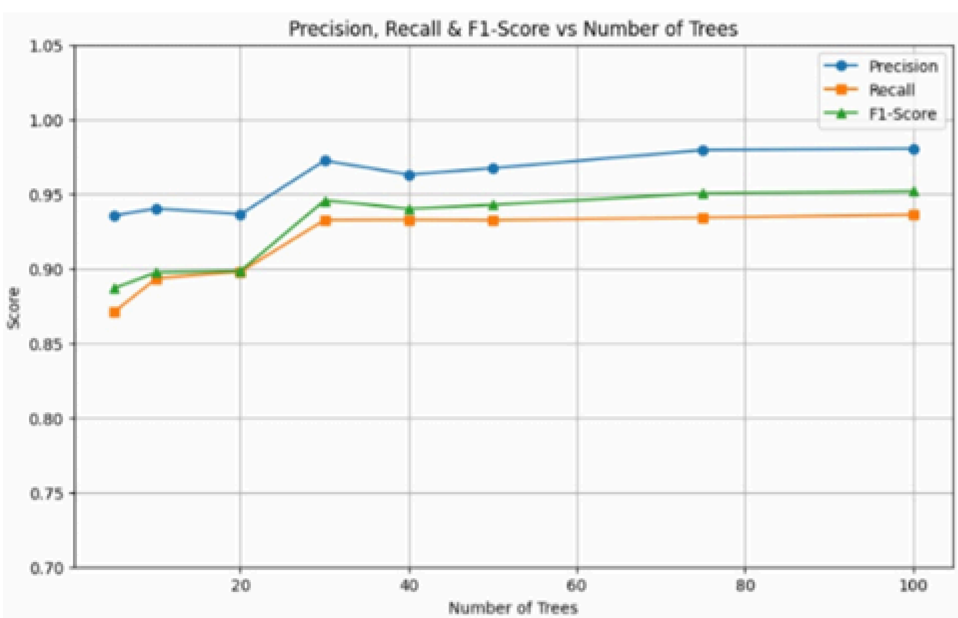


Fig. 4. Precision, Recall & F1 Score vs Number of Trees

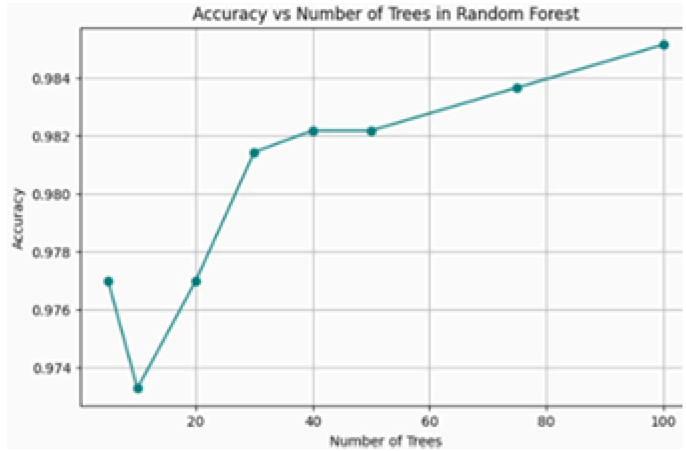


Fig. 5. Accuracy vs Number of Trees

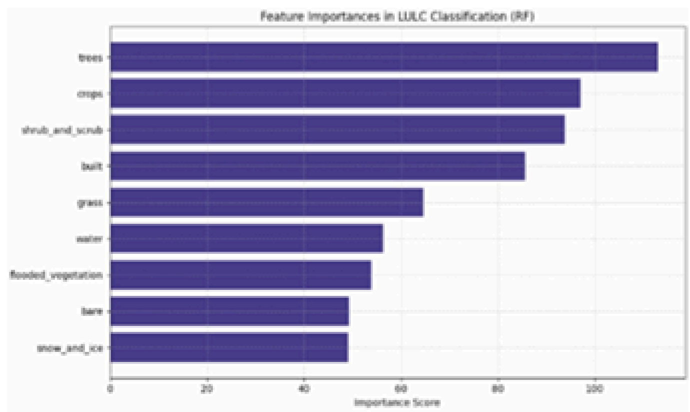


Fig. 6. Feature Importances in Random Forest Model

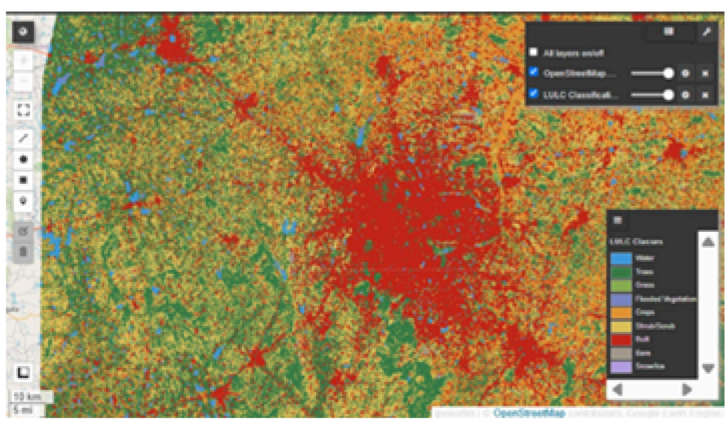


Fig. 7. LULC Classification Map using RF Model

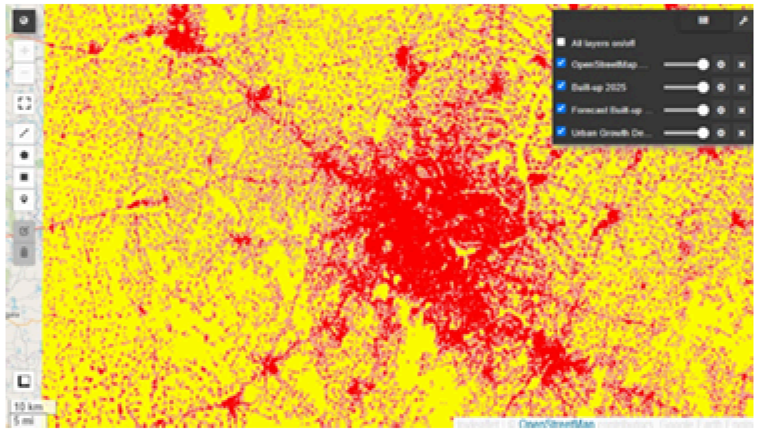


Fig. 8. Forecasted Built-up Area for 2026

Fig. 9. Projected Trees to Built-Up Transition (2025–2026)

Table 1. CA Evaluation Metrics.

Year Range	IoU Score	Tree → Built-up Area (sq.km)
2020-2021	0.8689	16.81
2023-2024	0.8638	29.77
2024-2025	0.8622	41.8

Table 2. Random Forest Evaluation Metrics.

Metric	Value	Notes
Overall Accuracy	0.9821	50 Trees
Kappa Coefficient	0.9752	Strong Agreement
Mean IoU	0.9004	High Class Balance

The proposed system demonstrated strong capability in classifying and forecasting land use and land cover (LULC) patterns over Bangalore using the Dynamic World dataset in conjunction with Sentinel-2 imagery. The Random Forest (RF) model, trained on 5000 sample points derived from band-wise class probabilities, effectively categorized land cover into nine distinct classes. These included vegetation types, built-up zones, and natural landforms. The classifier achieved a high overall accuracy of 98.21% and a Kappa coefficient of 0.975. These are indicative of strong agreement between predicted and actual labels (see Figure 3. RF Confusion Matrix).
Each class's performance was evaluated using Intersection over Union (IoU), with built-up areas reaching an IoU of 0.9855, and trees achieving 0.9765. The mean IoU across all classes stood at 0.9004, suggesting balanced model performance across diverse landscapes (see Table 2. Evaluation Metrics for Random Forest Classification).

To assess model robustness, metrics such as precision, recall, and F1-score were calculated for varying numbers of trees in the Random Forest. The analysis confirmed that 50 trees provided the best balance between accuracy and computational efficiency. The trends are plotted in Figure 4. Precision, Recall & F1 Score vs Number of Trees, with accuracy trends illustrated separately in Figure 5. Accuracy vs Number of Trees.

Feature importance analysis revealed that the ‘trees’ class was the most influential, followed by ‘crops’ and ‘shrub_and_scrub’, as shown in Figure 6. Feature Importances in Random Forest Model. This highlights the significant role of vegetative land in informing urban classification decisions.
The classification output for the test region is visualized in Figure 7. LULC Classification Map using RF Model, confirming a coherent spatial distribution of land use types in the region.

For forecasting, a Cellular Automata (CA)-based growth model was employed.
The 2025 built-up area was extracted using the mode of Dynamic World labels, and the CA simulation projected likely expansion for 2026 using a 3x3 kernel neighborhood. The forecasted built-up area for 2026 was estimated at 1574.93 sq.km, compared to 1482.86 sq.km in 2025—marking a net increase of 92.07 sq.km and a growth rate of 6.21% (see Figure 8. Forecasted Built-up Area for 2026 and Table 1. Evaluation Metrics for CA-Based Urban Growth Forecasting).
To evaluate CA reliability, backcasting analysis was conducted for 2020→2021, 2023→2024, and 2024→2025. IoU scores of 0.8689, 0.8638, and 0.8622, respectively, validated the temporal coherence of the simulated growth (see Table 1).
One of the critical environmental insights of this study involves tree loss

estimation. A focused simulation assessed tree-covered pixels in 2025 expected to convert to urban zones in 2026, resulting in a projected loss of 41.80 sq.km (see Figure 9. Projected Trees to Built-Up Transition (2025–2026)).

These results align closely with actual trends. For instance, between 2023 and 2024, 41.45 sq.km of tree land was observed to have been converted to built-up areas, with 16.81 sq.km between 2020 and 2021. CA simulation for 2023→2024 predicted 29.77 sq.km of tree loss, reaffirming the reliability of our approach (see Figure 10). Simulated Trees to Built-Up Transition (2023–2024)).

The study underlines the dual utility of this system: a robust LULC classifier powered by a well-tuned Random Forest, and a lightweight yet interpretable urban forecasting tool using Cellular Automata. While the CA model does not account for socio-political constraints or economic drivers, its ability to spatially simulate plausible urban growth offers a valuable decision-support resource for planners and sustainability advocates.

8 CONCLUSION AND FUTURE WORK

This project successfully demonstrates a hybrid framework that combines machine learning and spatial simulation for short-term urban growth forecasting and land use/land cover (LULC) classification. Using Google's Dynamic World V1 dataset, which provides per-pixel class probabilities across nine LULC categories, the system delivers interpretable, high-resolution insights into land dynamics in the Bangalore metropolitan region.

In the classification stage, a Random Forest (RF) model was trained on the probability bands derived from early 2025 satellite data. The model achieved a high overall accuracy of 98.21%, with a Kappa coefficient of 0.9752 and a mean Intersection over Union (IoU) of 0.9004 across the nine classes. The use of class probability bands, rather than static LULC labels, allowed the model to handle uncertain or mixed land types more effectively—especially in transition zones between vegetation and built-up land. Feature importance analysis further revealed that certain classes, such as built-up and trees, had significant weight in classification decisions, improving model transparency and explainability.

The forecasting stage employed a Cellular Automata (CA) model using a 3×3 neighborhood kernel to simulate 2026 urban expansion from 2025 built-up clusters. This rule-based simulation reflects the natural process of outward urban sprawl, marking adjacent non-built-up pixels as potential growth zones. The CA model predicted a 6.21% increase in built-up area within a year, and identified 41.8 sq.km of tree cover at risk of conversion to urban land. Backcasting exercises for the years 2020–2021 and 2023–2024 yielded IoU scores above 0.86, validating the reliability of the spatial growth logic. These predictions offer meaningful insights for land planners and urban governance bodies focused on managing ecological loss and controlling unplanned expansion.

All modeling and analysis were executed within the Google Earth Engine (GEE) environment using Python-based libraries such as geemap, Earth Engine API, NumPy, and matplotlib. The system supports sustainable development goals (SDGs), particularly those targeting sustainable cities (SDG 11), climate action (SDG 13), and life on land (SDG 15), by providing an accessible, interpretable, and high-resolution monitoring tool for urban growth and environmental impact.

Despite its achievements, the current project has limitations. The CA model is driven only by spatial proximity and does not incorporate other real-world drivers of urbanization, such as proximity to roads, economic zones, or demographic patterns. Additionally, the simulation operates in a single time step and does not account for temporal probability transitions as seen in CA-Markov models. There is also no integration of socio-economic datasets, zoning maps, or real-time policy inputs.

Future work will focus on enhancing the simulation engine by introducing multi-factor CA models that combine road networks, elevation, and land value with spatial rules. Integration of temporal transition matrices (e.g., CA-Markov) can improve multi-year forecasting. A deep learning–based classification model such as U-Net or SegFormer can be experimented with for improved accuracy. The system can be extended into an interactive dashboard using tools like Streamlit, allowing policymakers to visualize growth under different zoning or development scenarios. This will transform the project into a robust, real-world decision support system for urban planning.

9 REFERENCES

1. Kamusoko, C., Gamba, J.: Simulating Urban Growth Using a Random Forest-Cellular Automata (RF-CA) Model. In: International Journal of Geo-Information, vol. 4(2), pp. 447–470. MDPI, Basel (2015).
2. Brown, C.F., Tuholske, C., Tellman, B., Evans, T.P., Wentz, E.A., Stokes, E.: Dynamic World, Near Real-Time Global 10 m Land Use Land Cover Mapping. In: Scientific Data, vol. 9, article 687. Nature Portfolio (2022).
3. Thottolil, J., Neelakantan, A., Anjana, N.M., Ganapathy, S.: Forecasting Urban Expansion of Bangalore Using CA-PLUS Model. In: Preprints.org (2023).
4. Raj, S., Gowtham, B.G.: Modelling and Prediction of Urban Growth Using SLEUTH Model – A Case Study of Mysuru City, India. In: IOSR Journal of Humanities and Social Science (IOSR-JHSS), vol. 29(1), Ser. 5, pp. 25–35. IOSR Journals (2024).
5. Yao, Y., Liu, X., Zhang, D., Liang, Z., Zhang, Y.: Simulation of Urban Expansion and Farmland Loss in China by Integrating Cellular Automata and Random Forest. In: arXiv preprint arXiv:1705.05651 (2017).
6. Zhang, Y., Li, X., Liu, X., Chen, Y.: Forecasting Urban Land Use Change Based on Cellular Automata and the PLUS Model. In: Land, vol. 11(5), article 652. MDPI, Basel (2022).
7. Girma, R., Fürst, C., Moges, A.: Land Use Land Cover Change Modeling by Integrating Artificial Neural Network with Cellular Automata-Markov Chain Model in Gidabo River Basin, Main Ethiopian Rift. In: Environmental Challenges, vol. 6, article 100419. Elsevier (2022).
8. Gidey, E., Dikinya, O., Sebego, R., Segosebe, E., Zenebe, A.: Cellular Automata and Markov Chain (CA_Markov) Model-Based Predictions of Future Land Use and Land Cover Scenarios (2015–2033) in Raya, Northern Ethiopia. In: Modeling Earth Systems and Environment, vol. 3, pp. 1245–1262. Springer (2017).
9. Fathizad, H., Rostami, N., Faramarzi, M.: Detection and Prediction of Land Cover Changes Using Markov Chain Model in Semi-Arid Rangeland in Western Iran. In: Environmental Monitoring and Assessment, vol. 187, article 629. Springer (2015).
10. Gharaibeh, A., Shaamala, A., Obeidat, R., Al-Kofahi, S.: Improving Land-Use Change Modeling by Integrating ANN with Cellular Automata-Markov Chain Model. In: Heliyon, vol. 6(5), article e05092. Elsevier (2020).
11. Habibie, M.I.: The Application of Machine Learning Using Google Earth Engine for Remote Sensing Analysis. In: Jurnal Teknologi Informasi, vol. 16(2), pp. 233–240. Universitas Muhammadiyah Surakarta (2022).
12. Gashaw, T., Bantider, A., Mahari, A.: Evaluations of Land Use/Land Cover Changes and Land Degradation in Dera District, Ethiopia: GIS and Remote Sensing Based Analysis. In: International Journal of Scientific Research in Environmental Sciences, vol. 2(5), pp. 199–208. International Scholars Journals (2014).
13. Hassan, Z., Shabbir, R., Ahmad, S.S., Malik, A.H., Aziz, N., Butt, A., Erum, S.: Dynamics of Land Use and Land Cover Change (LULCC) Using Geospatial Techniques: A Case Study of Islamabad, Pakistan. In: SpringerPlus, vol. 5, article 812. Springer (2016).
14. Gashaw, T., Tulu, T., Argaw, M., Worqlul, A.W.: Modeling the Impacts of Land Use/Land Cover Changes on the Hydrology of the Andassa Watershed, Blue Nile Basin, Ethiopia. In: Science of the Total Environment, vol. 619–620, pp. 1394–1408. Elsevier (2018).

10 APPENDIX (SOURCE CODE)

```
# -*- coding: utf-8 -*-
"""capstoneproject25.ipynb

Automatically generated by Colab.

Original file is located at
    https://colab.research.google.com/drive/13O_h2Uhv1D0VhKGlq2eDpdXxVnOkP9v9
"""

import ee

ee.Authenticate()
ee.Initialize(project='majorproject-441517')

# Install Earth Engine and other dependencies
!pip install geemap earthengine-api -q

import geemap

# Define Bangalore ROI (Polygon covering the city)
bangalore_roi = ee.Geometry.Polygon([
    [[77.3, 13.0], [77.8, 13.0], [77.8, 12.7], [77.3, 12.7], [77.3, 13.0]]
])

# Define date range
START_DATE = ee.Date('2025-01-01')
END_DATE = ee.Date('2025-04-01')

# Filter Dynamic World dataset for Bangalore
dw_collection = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1') \
    .filterBounds(bangalore_roi) \
    .filterDate(START_DATE, END_DATE)

# Filter Sentinel-2 data
s2_collection = ee.ImageCollection('COPERNICUS/S2_HARMONIZED') \
    .filterBounds(bangalore_roi) \
    .filterDate(START_DATE, END_DATE)

# Link DW and S2 images
linked_col = dw_collection.linkCollection(s2_collection, s2_collection.first().bandNames())

# Get the first available linked image
linked_image = ee.Image(linked_col.first())

# Print available bands
```

```
print("Bands in Dynamic World dataset:", linked_image.bandNames().getInfo())

# Define LULC class names
CLASS_NAMES = [
    'water', 'trees', 'grass', 'flooded_vegetation', 'crops',
    'shrub_and_scrub', 'built', 'bare', 'snow_and_ice'
]

# Define visualization colors
VIS_PALETTE = [
    '419bdf', '397d49', '88b053', '7a87c6', 'e49635',
    'dfc35a', 'c4281b', 'a59b8f', 'b39fe1'
]

# Create an RGB visualization of the label (most likely class)
dw_rgb = (
    linked_image.select('label')
    .visualize(min=0, max=8, palette=VIS_PALETTE)
    .divide(255)  # Normalize for display
)

# Get the most likely class probability
top1_prob = linked_image.select(CLASS_NAMES).reduce(ee.Reducer.max())

# Create a hillshade effect based on probability
top1_prob_hillshade = ee.Terrain.hillshade(top1_prob.multiply(100)).divide(255)

# Blend the RGB image with the probability hillshade
dw_rgb_hillshade = dw_rgb.multiply(top1_prob_hillshade)

# Define Bangalore ROI (Use a proper polygon to avoid FAO GAUL collection)
bangalore_roi = ee.Geometry.Polygon([
    [[77.3, 13.5], [77.8, 13.5], [77.8, 12.8], [77.3, 12.8], [77.3, 13.5]]
])

# Filter Dynamic World dataset for Jan 1 - April 1, 2025
dw_collection = (ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1')
                 .filterBounds(bangalore_roi)
                 .filterDate('2025-01-01', '2025-04-01')
                 .select('label'))  # Select only the LULC classification band

# Create a mode composite (most frequent class over time) to reduce noise
dw_mode = dw_collection.reduce(ee.Reducer.mode())

# Define visualization parameters
class_palette = [
    '419bdf', '397d49', '88b053', '7a87c6', 'e49635',
    'dfc35a', 'c4281b', 'a59b8f', 'b39fe1'
```

```
]
vis_params = {
    'min': 0, 'max': 8, 'palette': class_palette
}

# Create an interactive map
m = geemap.Map(center=[13.0, 77.5], zoom=10)
m.addLayer(dw_mode, vis_params, "LULC Classification (Fixed)")

# Show the map
m

# Define feature bands (probabilities of each land cover class)
FEATURE_BANDS = ['water', 'trees', 'grass', 'flooded_vegetation', 'crops',
                 'shrub_and_scrub', 'built', 'bare', 'snow_and_ice']

# Label band (ground truth)
LABEL_BAND = 'label'

# Select feature bands and label from Dynamic World image
dw_features = linked_image.select(FEATURE_BANDS + [LABEL_BAND])

# Generate random sample points within the Bangalore ROI
training_samples = dw_features.sample(
    region=bangalore_roi,
    scale=10,  # Dynamic World resolution (10m)
    numPixels=5000,  # Total number of sample points
    seed=42,
    geometries=True  # Keep point geometry for visualization
)

# Print sample data
print("Training Samples:", training_samples.first().getInfo())

# Split data into 70% training and 30% testing
split = 0.7
training_set = training_samples.randomColumn('split').filter(ee.Filter.lt('split', split))
testing_set = training_samples.randomColumn('split').filter(ee.Filter.gte('split', split))

print(f"Training Samples: {training_set.size().getInfo()}")
print(f"Testing Samples: {testing_set.size().getInfo()}")

# Define Random Forest classifier with 50 trees
classifier = ee.Classifier.smileRandomForest(50).train(
    features=training_set,
    classProperty=LABEL_BAND,
    inputProperties=FEATURE_BANDS
)
```

```
# Apply classifier to the image
classified_image = dw_features.classify(classifier)

# Classify the test dataset
test_predictions = testing_set.classify(classifier)

# Generate an error matrix
error_matrix = test_predictions.errorMatrix(LABEL_BAND, 'classification')

# Print Accuracy Metrics
print("Confusion Matrix:\n", error_matrix.getInfo())
print("Overall Accuracy:", error_matrix.accuracy().getInfo())

import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np

# Get the confusion matrix from errorMatrix
conf_matrix = np.array(error_matrix.getInfo())

# Plot it using seaborn
plt.figure(figsize=(10, 8))
sns.heatmap(conf_matrix, annot=True, fmt='d', cmap='Blues',
        xticklabels=CLASS_NAMES, yticklabels=CLASS_NAMES)

plt.title("Random Forest Confusion Matrix")
plt.xlabel("Predicted Label")
plt.ylabel("True Label")
plt.xticks(rotation=45)
plt.yticks(rotation=0)
plt.tight_layout()
plt.show()

# Normalize rows (true classes)
conf_matrix_norm = conf_matrix.astype('float') / conf_matrix.sum(axis=1)[:, np.newaxis]

plt.figure(figsize=(10, 8))
sns.heatmap(conf_matrix_norm, annot=True, fmt=".2f", cmap='viridis',
        xticklabels=CLASS_NAMES, yticklabels=CLASS_NAMES)

plt.title("Normalized Confusion Matrix")
plt.xlabel("Predicted Label")
plt.ylabel("True Label")
plt.xticks(rotation=45)
plt.yticks(rotation=0)
plt.tight_layout()
plt.show()
```

```
# Normalize rows (true classes)
conf_matrix_norm = conf_matrix.astype('float') / conf_matrix.sum(axis=1)[:, np.newaxis]

plt.figure(figsize=(10, 8))
sns.heatmap(conf_matrix_norm, annot=True, fmt=".2f", cmap='viridis',
            xticklabels=CLASS_NAMES, yticklabels=CLASS_NAMES)

plt.title("Normalized Confusion Matrix")
plt.xlabel("Predicted Label")
plt.ylabel("True Label")
plt.xticks(rotation=45)
plt.yticks(rotation=0)
plt.tight_layout()
plt.show()

# Create an interactive map
m = geemap.Map(center=[13.0, 77.5], zoom=10)

# Add the classified LULC layer
m.addLayer(dw_mode, vis_params, "LULC Classification (Fixed)")

# Define dictionary of class names and their colors
legend_dict = {
    'Water': '#419bdf',
    'Trees': '#397d49',
    'Grass': '#88b053',
    'Flooded Vegetation': '#7a87c6',
    'Crops': '#e49635',
    'Shrub/Scrub': '#dfc35a',
    'Built': '#c4281b',
    'Bare': '#a59b8f',
    'Snow/Ice': '#b39fe1'
}

# Add the legend to the map
m.add_legend(title="LULC Classes", legend_dict=legend_dict)

# Show the map
m

print("Kappa Coefficient:", error_matrix.kappa().getInfo())

# Get the confusion matrix as a 2D list
conf_matrix = error_matrix.array().getInfo()

# Initialize list to hold IoU for each class
ious = []
```

```
# Number of classes
num_classes = len(conf_matrix)

for i in range(num_classes):
    TP = conf_matrix[i][i]
    FP = sum(conf_matrix[j][i] for j in range(num_classes)) - TP
    FN = sum(conf_matrix[i][j] for j in range(num_classes)) - TP
    denom = TP + FP + FN
    iou = TP / denom if denom != 0 else 0
    ious.append(iou)

# Print IoU for each class
print("\nIoU per Class:")
for i, iou in enumerate(ious):
    print(f"Class {i} ({CLASS_NAMES[i]}): IoU = {iou:.4f}")

# Optional: Mean IoU
mean_iou = sum(ious) / len(ious)
print(f"\nMean IoU: {mean_iou:.4f}")

import matplotlib.pyplot as plt

# Define range of tree counts to test
tree_counts = [5, 10, 20, 30, 40, 50, 75, 100]
accuracies = []

for n_trees in tree_counts:
    # Train Random Forest with n_trees
    classifier = ee.Classifier.smileRandomForest(n_trees).train(
        features=training_set,
        classProperty=LABEL_BAND,
        inputProperties=FEATURE_BANDS
    )

    # Classify test data
    predictions = testing_set.classify(classifier)

    # Get error matrix and accuracy
    error_matrix = predictions.errorMatrix(LABEL_BAND, 'classification')
    accuracy = error_matrix.accuracy().getInfo()
    accuracies.append(accuracy)
    print(f"Trees: {n_trees}, Accuracy: {accuracy:.4f}")

# Plotting the results
plt.figure(figsize=(8, 5))
plt.plot(tree_counts, accuracies, marker='o', linestyle='-', color='teal')
plt.title('Accuracy vs Number of Trees in Random Forest')
```

```
plt.xlabel('Number of Trees')
plt.ylabel('Accuracy')
plt.grid(True)
plt.show()

import numpy as np
import matplotlib.pyplot as plt

# Range of tree counts to evaluate
tree_counts = [5, 10, 20, 30, 40, 50, 75, 100]

# Initialize lists to store average metrics across all classes
avg_precision_list = []
avg_recall_list = []
avg_f1_list = []

for n_trees in tree_counts:
    # Train Random Forest model
    classifier = ee.Classifier.smileRandomForest(n_trees).train(
        features=training_set,
        classProperty=LABEL_BAND,
        inputProperties=FEATURE_BANDS
    )

    # Classify test data
    predictions = testing_set.classify(classifier)

    # Get confusion matrix as a numpy array
    matrix = np.array(predictions.errorMatrix(LABEL_BAND, 'classification').getInfo())

    # Precision, recall, F1 per class
    precision = []
    recall = []
    f1 = []

    for i in range(len(matrix)):
        tp = matrix[i][i]
        fp = sum(matrix[:, i]) - tp
        fn = sum(matrix[i, :]) - tp

        p = tp / (tp + fp) if (tp + fp) != 0 else 0
        r = tp / (tp + fn) if (tp + fn) != 0 else 0
        f1_score = (2 * p * r) / (p + r) if (p + r) != 0 else 0

        precision.append(p)
        recall.append(r)
        f1.append(f1_score)
```

```
    # Store average scores
    avg_precision_list.append(np.mean(precision))
    avg_recall_list.append(np.mean(recall))
    avg_f1_list.append(np.mean(f1))

    print(f"Trees: {n_trees} | Precision: {np.mean(precision):.4f}, Recall: {np.mean(recall):.4f}, F1:
{np.mean(f1):.4f}")

# Plot curves
plt.figure(figsize=(10, 6))
plt.plot(tree_counts, avg_precision_list, label='Precision', marker='o')
plt.plot(tree_counts, avg_recall_list, label='Recall', marker='s')
plt.plot(tree_counts, avg_f1_list, label='F1-Score', marker='^')

plt.title('Precision, Recall & F1-Score vs Number of Trees')
plt.xlabel('Number of Trees')
plt.ylabel('Score')
plt.ylim(0.7, 1.05)
plt.grid(True)
plt.legend()
plt.show()

# Get Dynamic World images from 2020 to 2025
dw_2020_2025 = (ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1')
            .filterBounds(bangalore_roi)
            .filterDate('2020-01-01', '2025-04-01')
            .select('label'))

# Reduce to mode to get dominant land cover per pixel
lulc_2020_2025 = dw_2020_2025.reduce(ee.Reducer.mode())

# Extract built-up area from 2020-2025
builtup_mask_2025 = lulc_2020_2025.eq(6)

# Neighborhood kernel (3x3)
kernel = ee.Kernel.circle(radius=1)  # 3x3 kernel

# Create growth potential: areas surrounding current built-up
potential_growth = builtup_mask_2025.focal_max(kernel=kernel).And(builtup_mask_2025.Not())

# Simulate 1 step urban growth: 2026 = existing + surrounding likely growth
projected_builtup_2026 = builtup_mask_2025.Or(potential_growth)

# Visualization params
vis = {'min': 0, 'max': 1, 'palette': ['yellow', 'red']}

# Create map
forecast_map = geemap.Map(center=[13.0, 77.5], zoom=10)
```

```
forecast_map.addLayer(builtup_mask_2025, vis, 'Built-up 2025')
forecast_map.addLayer(projected_builtup_2026, vis, 'Forecast Built-up 2026')
forecast_map

# Define pixel area (10m resolution = 100 sq.m)
pixel_area = ee.Image.pixelArea().divide(1e6)  # convert to sq.km

# Calculate area of built-up in 2025
area_2025 = builtup_mask_2025.multiply(pixel_area)
total_builtup_2025 = area_2025.reduceRegion(
    reducer=ee.Reducer.sum(),
    geometry=bangalore_roi,
    scale=10,
    maxPixels=1e9
).get('label_mode')  # since builtup_mask_2025 is from 'label_mode' image

# Calculate area of built-up in 2026
area_2026 = projected_builtup_2026.multiply(pixel_area)
total_builtup_2026 = area_2026.reduceRegion(
    reducer=ee.Reducer.sum(),
    geometry=bangalore_roi,
    scale=10,
    maxPixels=1e9
).get('label_mode')

print("Built-up Area in 2025 (sq.km):", total_builtup_2025.getInfo())
print("Built-up Area in 2026 (sq.km):", total_builtup_2026.getInfo())

increase = ee.Number(total_builtup_2026).subtract(total_builtup_2025)
percent_growth = increase.divide(total_builtup_2025).multiply(100)

print("Net Increase in Built-up Area (sq.km):", increase.getInfo())
print("Percentage Growth in Built-up Area:", percent_growth.getInfo(), "%")

# Urban growth = newly built areas
urban_growth = projected_builtup_2026.And(builtup_mask_2025.Not()).selfMask()

# Use focal sum to count built pixels in 5x5 window (roughly 50m radius)
density_kernel = ee.Kernel.square(radius=5)
urban_growth_density = urban_growth.convolve(density_kernel)

# Add to map
forecast_map.addLayer(urban_growth_density, {'min': 0, 'max': 25, 'palette': ['white', 'red']}, 'Urban Growth
Density')
forecast_map

# Step 1: Get 2020 built-up
dw_2020 = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1') \
```

```
    .filterBounds(bangalore_roi) \
    .filterDate('2020-01-01', '2020-12-31') \
    .select('label') \
    .reduce(ee.Reducer.mode())  # most common class

builtup_2020 = dw_2020.eq(6)

# Step 2: Get 2021 actual built-up
dw_2021 = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1') \
    .filterBounds(bangalore_roi) \
    .filterDate('2021-01-01', '2021-12-31') \
    .select('label') \
    .reduce(ee.Reducer.mode())

builtup_2021_actual = dw_2021.eq(6)

# Step 3: Simulate 2021 built-up using CA logic from 2020
kernel = ee.Kernel.circle(radius=1)
potential_growth_2021 = builtup_2020.focal_max(kernel=kernel).And(builtup_2020.Not())
builtup_2021_predicted = builtup_2020.Or(potential_growth_2021)

# Step 4: Compare Predicted vs Actual
intersection = builtup_2021_predicted.And(builtup_2021_actual)
union = builtup_2021_predicted.Or(builtup_2021_actual)

# Step 5: Calculate area-based IoU
pixel_area = ee.Image.pixelArea().divide(1e6)

intersection_area = intersection.multiply(pixel_area).reduceRegion(
    reducer=ee.Reducer.sum(),
    geometry=bangalore_roi,
    scale=10,
    maxPixels=1e9
).get('label_mode')

union_area = union.multiply(pixel_area).reduceRegion(
    reducer=ee.Reducer.sum(),
    geometry=bangalore_roi,
    scale=10,
    maxPixels=1e9
).get('label_mode')

iou = ee.Number(intersection_area).divide(union_area)

print("⚙️ Backcasting IoU (2020 → 2021 CA Forecast):", iou.getInfo())

# Step 1: Get 2023 built-up
dw_2023 = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1') \
```

```
    .filterBounds(bangalore_roi) \
    .filterDate('2023-01-01', '2023-12-31') \
    .select('label') \
    .reduce(ee.Reducer.mode())  # most common class

builtup_2023 = dw_2023.eq(6)

# Step 2: Get 2024 actual built-up
dw_2024 = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1') \
    .filterBounds(bangalore_roi) \
    .filterDate('2024-01-01', '2024-12-31') \
    .select('label') \
    .reduce(ee.Reducer.mode())

builtup_2024_actual = dw_2024.eq(6)

# Step 3: Simulate 2024 built-up using CA logic from 2023
kernel = ee.Kernel.circle(radius=1)
potential_growth_2024 = builtup_2023.focal_max(kernel=kernel).And(builtup_2023.Not())
builtup_2024_predicted = builtup_2023.Or(potential_growth_2024)

# Step 4: Compare Predicted vs Actual
intersection = builtup_2024_predicted.And(builtup_2024_actual)
union = builtup_2024_predicted.Or(builtup_2024_actual)

# Step 5: Calculate area-based IoU
pixel_area = ee.Image.pixelArea().divide(1e6)

intersection_area = intersection.multiply(pixel_area).reduceRegion(
    reducer=ee.Reducer.sum(),
    geometry=bangalore_roi,
    scale=10,
    maxPixels=1e9
).get('label_mode')

union_area = union.multiply(pixel_area).reduceRegion(
    reducer=ee.Reducer.sum(),
    geometry=bangalore_roi,
    scale=10,
    maxPixels=1e9
).get('label_mode')

iou = ee.Number(intersection_area).divide(union_area)

print("⚙️ Backcasting IoU (2023 → 2024 CA Forecast):", iou.getInfo())

# Train the RF model again and get importance of features
rf_with_importance = ee.Classifier.smileRandomForest(50).train(
```

```
    features=training_set,
    classProperty=LABEL_BAND,
    inputProperties=FEATURE_BANDS
)

# Get feature importances
importance = rf_with_importance.explain().get('importance')

# Convert to dictionary and display
importance_dict = importance.getInfo()
sorted_importance = dict(sorted(importance_dict.items(), key=lambda x: x[1], reverse=True))

# Print in order of impact
print("🔍 Feature Importances (RF):")
for k, v in sorted_importance.items():
    print(f"{k}: {v:.4f}")

import matplotlib.pyplot as plt

# Plot feature importance
features = list(sorted_importance.keys())
importances = list(sorted_importance.values())

plt.figure(figsize=(10, 6))
plt.barh(features[::-1], importances[::-1], color='darkslateblue')
plt.xlabel("Importance Score")
plt.title("Feature Importances in LULC Classification (RF)")
plt.grid(True, linestyle='--', alpha=0.5)
plt.tight_layout()
plt.show()

# Get LULC maps for 2020 and 2025
lulc_2020 = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1') \
    .filterBounds(bangalore_roi) \
    .filterDate('2020-01-01', '2020-12-31') \
    .select('label') \
    .reduce(ee.Reducer.mode())

lulc_2025 = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1') \
    .filterBounds(bangalore_roi) \
    .filterDate('2025-01-01', '2025-04-01') \
    .select('label') \
    .reduce(ee.Reducer.mode())

# Detect pixels where trees became built
trees_to_built = lulc_2020.eq(1).And(lulc_2025.eq(6))

# Visualize in red
```

```
m = geemap.Map(center=[13.0, 77.5], zoom=10)
m.addLayer(trees_to_built.updateMask(trees_to_built), {'palette': 'red'}, 'Trees → Built-up (2020–2025)')
m

# Mask for 'trees' in 2025
trees_2025 = lulc_2025.eq(1)

# CA step: identify tree pixels near built-up areas
# We'll look at tree pixels that are near built-up areas (growth candidates)
kernel = ee.Kernel.circle(radius=1)  # 3x3 neighborhood

# Get built-up mask from 2025
builtup_2025 = lulc_2025.eq(6)

# Focal max to expand built-up outward by one pixel
expanded_builtup = builtup_2025.focal_max(kernel=kernel)

# Trees that are *near* built-up
tree_growth_candidates = trees_2025.And(expanded_builtup)

# Simulated 2026: trees that might turn built-up
tree_to_built_2026 = tree_growth_candidates

# Visualize
tree_growth_map = geemap.Map(center=[13.0, 77.5], zoom=10)
tree_growth_map.addLayer(trees_2025.updateMask(trees_2025), {'palette': 'green'}, 'Trees 2025')
tree_growth_map.addLayer(tree_to_built_2026.updateMask(tree_to_built_2026), {'palette': 'red'}, 'CA
Forecast: Trees → Built-up 2026')
tree_growth_map

# Step 1: Mask for trees in 2025
trees_2025 = lulc_2025.eq(1)

# Step 2: Use your CA-based forecast of 2026 built-up (already computed)
# 'projected_builtup_2026' from previous CA logic

# Step 3: Identify areas that were trees in 2025 but projected to become built-up in 2026
ca_tree_to_built_2026 = trees_2025.And(projected_builtup_2026)

# Step 4: Calculate pixel area (in sq.km)
pixel_area = ee.Image.pixelArea().divide(1e6)

# Step 5: Multiply mask by pixel area
tree_loss_area_image = ca_tree_to_built_2026.multiply(pixel_area)

# Step 6: Reduce region to get total area
tree_loss_area = tree_loss_area_image.reduceRegion(
    reducer=ee.Reducer.sum(),
```

```
    geometry=bangalore_roi,
    scale=10,
    maxPixels=1e9
).get('label_mode')  # because CA mask was from lulc_2025

# Step 7: Print result
print("CA Projected Tree → Built-up Area Loss (2025–2026) in sq.km:", tree_loss_area.getInfo())

# Get LULC maps for 2023 and 2024
lulc_2023 = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1') \
    .filterBounds(bangalore_roi) \
    .filterDate('2023-01-01', '2023-12-31') \
    .select('label') \
    .reduce(ee.Reducer.mode())

lulc_2024 = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1') \
    .filterBounds(bangalore_roi) \
    .filterDate('2024-01-01', '2024-12-31') \
    .select('label') \
    .reduce(ee.Reducer.mode())

# From 2023 to 2024: Trees → Built-up
tree_to_built_2024 = lulc_2023.eq(1).And(lulc_2024.eq(6))

# Estimate area (sq.km)
tree_loss_area_2024 = tree_to_built_2024.multiply(pixel_area).reduceRegion(
    reducer=ee.Reducer.sum(),
    geometry=bangalore_roi,
    scale=10,
    maxPixels=1e9
).get('label_mode')

# Print result
print("Actual Tree → Built-up Area (2023–2024):", tree_loss_area_2024.getInfo(), "sq.km")

# Get LULC maps for 2020 and 2021
lulc_2020 = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1') \
    .filterBounds(bangalore_roi) \
    .filterDate('2020-01-01', '2020-12-31') \
    .select('label') \
    .reduce(ee.Reducer.mode())

lulc_2021 = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1') \
    .filterBounds(bangalore_roi) \
    .filterDate('2021-01-01', '2021-12-31') \
    .select('label') \
    .reduce(ee.Reducer.mode())
```

```
# From 2023 to 2024: Trees → Built-up
tree_to_built_2021 = lulc_2020.eq(1).And(lulc_2021.eq(6))

# Estimate area (sq.km)
tree_loss_area_2021 = tree_to_built_2021.multiply(pixel_area).reduceRegion(
    reducer=ee.Reducer.sum(),
    geometry=bangalore_roi,
    scale=10,
    maxPixels=1e9
).get('label_mode')

# Print result
print("Actual Tree → Built-up Area (2020–2021):", tree_loss_area_2021.getInfo(), "sq.km")

# Get LULC maps for 2023 and 2024
lulc_2023 = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1') \
    .filterBounds(bangalore_roi) \
    .filterDate('2023-01-01', '2023-12-31') \
    .select('label') \
    .reduce(ee.Reducer.mode())

lulc_2024 = ee.ImageCollection('GOOGLE/DYNAMICWORLD/V1') \
    .filterBounds(bangalore_roi) \
    .filterDate('2024-01-01', '2024-12-31') \
    .select('label') \
    .reduce(ee.Reducer.mode())

# Built-up mask from 2023
builtup_mask_2023 = lulc_2023.eq(6)

# 3x3 neighborhood kernel
kernel = ee.Kernel.circle(radius=1)

# Potential growth = neighbors of built-up that are not built-up
potential_growth_2024 = builtup_mask_2023.focal_max(kernel=kernel).And(builtup_mask_2023.Not())

# Simulate 2024 growth
projected_builtup_2024 = builtup_mask_2023.Or(potential_growth_2024)

# Find where trees existed in 2023 and turned into built-up (via simulation)
ca_tree_to_built = lulc_2023.eq(1).And(projected_builtup_2024)

# Calculate area of this conversion
pixel_area = ee.Image.pixelArea().divide(1e6)  # convert to sq.km
tree_to_built_area = ca_tree_to_built.multiply(pixel_area)

# Sum over Bangalore ROI
total_tree_to_built_area = tree_to_built_area.reduceRegion(
```

```
    reducer=ee.Reducer.sum(),
    geometry=bangalore_roi,
    scale=10,
    maxPixels=1e9
).get('label_mode')

# Display result
print("CA Projected Tree → Built-up Area Loss (2023–2024) in sq.km:", total_tree_to_built_area.getInfo())

# Visualize the conversion on the map
m = geemap.Map(center=[13.0, 77.5], zoom=10)
m.addLayer(ca_tree_to_built.updateMask(ca_tree_to_built), {'palette': 'red'}, 'CA Tree → Built-up
(2023–2024)')
m
```

GITHUB LINK

https://github.com/shrutinigamdsu/Team33_UrbanForesight

DAYANANDA SAGAR UNIVERSITY

Deverakaggalahalli, Harohalli, Kanakapura Rd, Dist. Ramanagara, Karnataka-562112

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

CERTIFICATE
OF ACHIEVEMENT

THIS CERTIFICATE IS PRESENTED TO :

SHRUTI NIGAM

For outstanding performance and excellence in the **Computer Vision & Machine Learning** category at the "Tech Spark 2.0" event, organized by the Department of Computer Science and Engineering (Artificial Intelligence and Machine Learning) held on 26th April 2025.

Dr. Jayavrinda Vrindavanam
Professor & Chairperson
DEPARTMENT OF CSE(AI & ML)
SOE, DSU

Dr. Udaya Kumar Reddy K.R.
Dean
SCHOOL OF ENGINEERING
DSU

DAYANANDA SAGAR UNIVERSITY

Deverakaggalahalli, Harohalli, Kanakapura Rd, Dist. Ramanagara, Karnataka-562112

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

CERTIFICATE

OF ACHIEVEMENT

THIS CERTIFICATE IS PRESENTED TO :

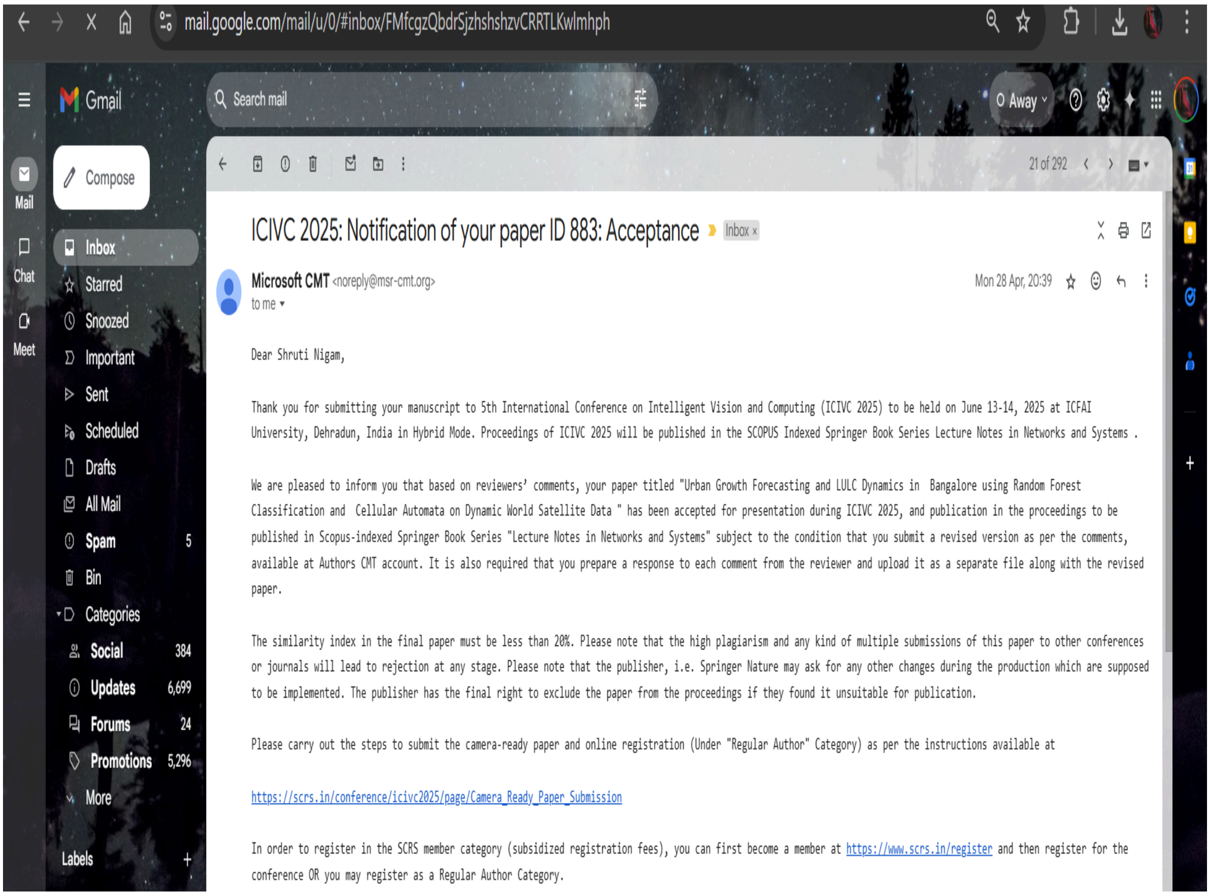
RISHITHA KATTIPALLEM

For outstanding performance and excellence in the **Computer Vision & Machine Learning** category at the "Tech Spark 2.0" event, organized by the Department of Computer Science and Engineering (Artificial Intelligence and Machine Learning) held on 26th April 2025.

Dr. Jayavrinda Vrindavanam
Professor & Chairperson
DEPARTMENT OF CSE(AI & ML)
SOE, DSU

Dr. Udaya Kumar Reddy K.R.
Dean
SCHOOL OF ENGINEERING
DSU

PAPER ACCEPTANCE EMAIL



ICIVC 2025: Notification of your paper ID 883: Acceptance

Microsoft CMT <noreply@msr-cmt.org>

to me

Mon 28 Apr, 20:39

Dear Shruti Nigam,

Thank you for submitting your manuscript to 5th International Conference on Intelligent Vision and Computing (ICIVC 2025) to be held on June 13-14, 2025 at ICFAI University, Dehradun, India in Hybrid Mode. Proceedings of ICIVC 2025 will be published in the SCOPUS Indexed Springer Book Series Lecture Notes in Networks and Systems .

We are pleased to inform you that based on reviewers' comments, your paper titled "Urban Growth Forecasting and LULC Dynamics in Bangalore using Random Forest Classification and Cellular Automata on Dynamic World Satellite Data " has been accepted for presentation during ICIVC 2025, and publication in the proceedings to be published in Scopus-indexed Springer Book Series "Lecture Notes in Networks and Systems" subject to the condition that you submit a revised version as per the comments, available at Authors CMT account. It is also required that you prepare a response to each comment from the reviewer and upload it as a separate file along with the revised paper.

The similarity index in the final paper must be less than 20%. Please note that the high plagiarism and any kind of multiple submissions of this paper to other conferences or journals will lead to rejection at any stage. Please note that the publisher, i.e. Springer Nature may ask for any other changes during the production which are supposed to be implemented. The publisher has the final right to exclude the paper from the proceedings if they found it unsuitable for publication.

Please carry out the steps to submit the camera-ready paper and online registration (Under "Regular Author" Category) as per the instructions available at

https://scrs.in/conference/icivc2025/page/Camera_Ready_Paper_Submission

In order to register in the SCRS member category (subsidized registration fees), you can first become a member at https://www.scrs.in/register and then register for the conference OR you may register as a Regular Author Category.

Urban Growth Forecasting and LULC Dynamics in Bangalore using Random Forest Classification and Cellular Automata on Dynamic World Satellite Data

Shruti Nigam[1], Rishitha Kattipallem[2], Bhuvana Mohini T N[3], Rangaraj BS[4]

[1] Dayananda Sagar University, Bangalore, Karnataka, India
shrutin567@gmail.com
[2] Dayananda Sagar University, Bangalore, Karnataka, India
Kattipallemrishitha315@gmail.com
[3] Dayananda Sagar University, Bangalore, Karnataka, India
bhuvana.m-aiml@dsu.edu.in
[4] Dayananda Sagar University, Bangalore, Karnataka, India
rangaraj-cse@dsu.edu.in

Abstract. Urbanization represents one of the most impactful land use and land cover (LULC) changes in rapidly expanding cities, especially in developing countries. Bangalore, India, has experienced substantial urban sprawl in recent years, often replacing vegetated areas. Forecasting built-up growth accurately is vital for sustainable planning and environmental management. Existing models often depend solely on historical spatial patterns, lacking integration of spatial dynamics with machine learning. This study proposes a hybrid framework that merges Random Forest (RF)-based classification with Cellular Automata (CA) for spatial growth simulation. Using Dynamic World (DW) Sentinel-2 imagery from 2020 to 2025, the RF classifier is trained on class probability bands and evaluated using confusion matrix, Kappa, and mean IoU. Built-up expansion for 2026 is simulated using CA-based neighborhood logic on 2025 built-up clusters. The system also detects tree-to-built-up conversions, offering environmental insights. The implementation uses Google Earth Engine and Python libraries like geemap and seaborn. Achieving over 98% accuracy, the model predicts a 6.2% urban growth by 2026, supporting data-driven urban planning and land use monitoring.

Keywords: Land Use and Land Cover (LULC), Random Forest Classification, Cellular Automata Urban Forecasting.

1 Introduction

Land Use and Land Cover (LULC) classification and forecasting are essential for monitoring urbanization, ecological change, and resource planning. In rapidly expanding cities like Bangalore, understanding land transitions—especially the growth of built-up areas—is vital for sustainable development and environmental management.

This study addresses two challenges: (1) accurate classification of current land cover using satellite imagery, and (2) forecasting future urban expansion through spatial modeling. While machine learning has improved LULC classification, predicting short-term land cover change remains difficult in heterogeneous urban areas, where vegetation, agriculture, and built-up zones shift rapidly. There is a pressing need for systems that not only classify land cover precisely but also model future changes with spatial context.

Open-access datasets like Sentinel-2 and Dynamic World, along with cloud platforms like Google Earth Engine (GEE), now support scalable analysis of spatiotemporal imagery. Random Forest classifiers have proven effective for LULC classification, while Cellular Automata (CA) offer spatially intuitive modeling of land transformation based on neighborhood dynamics.

Despite these advances, forecasting urban growth faces issues like intra-class variation and subtle transition zones. These challenges call for models that blend data-driven learning with interpretable spatial logic.

To address this, our work combines a Random Forest classifier (trained on Dynamic World class probabilities) with CA-based simulation to project built-up expansion for 2026, using patterns from 2020–2025. Implemented entirely on GEE, this approach also highlights environmental concerns by measuring green-to-urban transitions, particularly tree cover loss.

Our goal is to deliver a reproducible, interpretable, and scalable forecasting framework that supports informed urban planning, ecological monitoring, and sustainable development. The upcoming sections review related work, describe our methodology, present findings, and explore the real-world impact of our approach.

2 Literature Review

Urban growth and land use/land cover (LULC) classification have seen a range of approaches integrating machine learning and spatial dynamics. This section summarizes relevant works and their connection to our study.

[1] A hybrid RF-CA model was used to simulate urban expansion in Harare, Zimbabwe, where Random Forest generated transition potential maps that fed into a Cellular Automata model. The approach outperformed SVM-CA and LR-CA baselines, achieving a Kappa simulation accuracy of 0.51, highlighting the advantage of combining statistical and spatial modeling.

[2] The Dynamic World dataset, developed by Google and WRI, provides 10-meter resolution global LULC data using deep learning on Sentinel-2 imagery. It offers class probabilities across nine categories, making it well-suited for real-time urban monitoring and change detection tasks.

[3] In Bangalore, the CA-PLUS model was used to project urban growth up to 2030, integrating LULC data and socio-economic drivers. The study underlined the value of combining environmental and anthropogenic factors for enhanced urban forecasting accuracy.

[4] A study in Mysuru applied the SLEUTH CA-based model using Landsat data (1995–2022) and achieved 86.44% accuracy ($R^2 = 0.98$). This confirmed the efficacy of CA-based models in capturing temporal urban dynamics for regional planning.

Building on these efforts, our work applies the RF-CA framework with Dynamic World data to simulate urban growth in Bangalore for 2026. The integration of high-resolution, real-time LULC inputs with data-driven transition modeling addresses gaps in both accuracy and update frequency, especially critical for fast-developing regions.

3 Methodology

3.1 Problem Statement

The goal of this study is to develop an intelligent and interpretable land use and land cover (LULC) classification and forecasting system for the Bangalore region. The system aims to identify different land cover types using high-resolution satellite data and accurately forecast urban expansion patterns—specifically built-up area growth—by 2026.

To achieve this, the system integrates: A supervised machine learning model (Random Forest) for pixel-wise classification of Dynamic World Sentinel-2 data, A Cellular Automata (CA)-based neighborhood expansion approach for simulating future urban growth based on spatial patterns observed from 2020 to 2025.

The solution should ensure: Effective classification of land cover categories (e.g., trees, water, built-up, etc.) using multi-band input features, Forecasting of built-up area growth using spatial transition rules that mimic real-world urban sprawl tendencies, Quantitative assessment of model performance through accuracy, confusion matrix, Kappa score, Intersection over Union (IoU), and backcasting validations, Environmental insight by quantifying tree-to-built-up conversions and spatial density of predicted growth.

The system uses the Google Earth Engine (GEE) cloud platform and Dynamic World V1 dataset for near real-time land cover classification. The primary objective is to build a lightweight, explainable, and scalable forecasting model suitable for aiding city planners and researchers concerned with sustainable urban expansion.

System Architecture

Figure 1 illustrates the complete architecture of the proposed LULC classification and urban growth forecasting system. The system is built in two main stages: LULC Classification Module, Urban Growth Forecasting Module (Cellular Automata-Based).

Each stage works on top of satellite imagery and geospatial data from Google Earth Engine (GEE), using machine learning and spatial modeling to produce accurate land cover insights and simulate urban expansion.

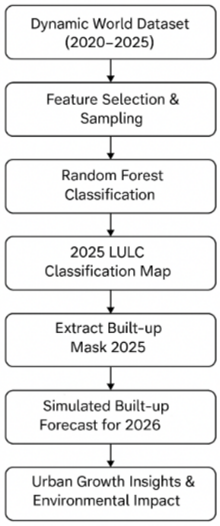


Fig. 1. System Architecture of the Proposed Model

The two-stage process begins with LULC (Land Use Land Cover) classification, where Sentinel-2 based Dynamic World images (2020–2025) for the Bangalore region are used as input. Probability bands for all land cover classes and the top-1 label band are extracted as features. Labeled training and testing points are generated through random sampling, and a Random Forest classifier is trained to classify each pixel into one of the nine Dynamic World land cover classes. Model performance is evaluated using metrics such as accuracy, Kappa coefficient, confusion matrix, and IoU, resulting in a classified LULC map for early 2025. In the second stage, urban growth forecasting is performed using a built-up mask derived from the 2020–2025 LULC classification. A Cellular Automata (CA)-based neighborhood expansion rule is applied, where pixels adjacent to existing built-up areas are marked as potential growth zones. This leads to the generation of a projected built-up map for 2026, followed by visualization and quantification of changes such as built-up area increase, tree-to-built conversions, and growth density heatmaps.

Datasets

To model and forecast urban expansion using machine learning and spatial dynamics, this study relies on high-resolution, temporally consistent datasets from Google Earth Engine (GEE), specifically the Dynamic World (DW) and Sentinel-2 Harmonized datasets. The DW dataset, developed by Google and the World Resources Institute, provides near real-time 10-meter resolution land use/land cover (LULC) classifications with probabilistic estimates across nine major categories (e.g., trees, built-up, crops, water). Its "label" band, which identifies the most likely class, serves as a key input for supervised classification and temporal analysis. The dataset is central to our workflow—used for feature extraction, Random Forest (RF) training, and change detection.

Complementing this, Sentinel-2 Harmonized imagery contributes spectral bands for visual validation, cloud masking, and additional quality checks. Though DW is derived from Sentinel-2, direct access to raw bands enriches classifier training and visualization accuracy. Data is filtered to the Bangalore region from 2020 to 2025, and random sampling via GEE's sample() function generates labeled points. These are split 70:30 into training and testing sets for performance evaluation using accuracy metrics and confusion matrices.

Despite its strengths, DW's per-pixel classification may introduce noise in heterogeneous zones, and its use of maximum probability labels can oversimplify mixed-use areas. However, the dataset's temporal granularity and spatial consistency make it highly suitable for LULC modeling. Together, DW and Sentinel-2 form a robust data foundation for simulating urban growth and assessing land transition dynamics over time.

4 Experimental Work

This study combines machine learning classification with Cellular Automata (CA)-based forecasting to model future land use and land cover (LULC) changes, focusing on urban expansion in Bangalore. The entire pipeline is implemented on Google Earth Engine (GEE) using Python and Geemap, ensuring spatial precision and scalability for urban sustainability research.

We utilize the Dynamic World (DW) dataset, which offers 10-meter resolution land cover data from January 2020 to April 2025, containing probabilistic estimates for nine LULC classes along with a "label" band indicating the most likely class. Sentinel-2 Harmonized imagery supports visual inspection and data quality checks. Data is spatially filtered to the Bangalore region, and relevant bands are selected.

Feature vectors are generated using the DW class probabilities, with the "label" band as ground truth. A random sampling approach is used to extract 5000 points, which are split into 70% training and 30% testing subsets. A Random Forest (RF) classifier with 50 trees is trained to predict LULC classes. Classification performance is evaluated using accuracy, confusion matrix, precision, recall, F1-score, and Intersection over Union (IoU). The model achieves over 98% accuracy, with high IoU values, particularly for built-up and trees. Feature importance analysis reveals that “trees,” “crops,” and “shrub_and_scrub” are most predictive.

Forecasting is performed using a simple yet effective CA model. A 3x3 kernel is used to identify potential growth zones adjacent to existing built-up areas using focal_max(). The 2026 map is generated by merging current built-up zones with these likely expansion pixels. This neighborhood-based logic simulates short-term urban growth where planning data is unavailable.

The complete pipeline—data acquisition, classification, and forecasting—is illustrated in Figure 2. The system also detects meaningful transitions such as "trees to built-up," offering valuable inputs for sustainability planning. Metrics like projected built-up growth and vegetation loss align with Sustainable Development Goals (SDGs), highlighting the model’s broader utility in urban policy and environmental monitoring.

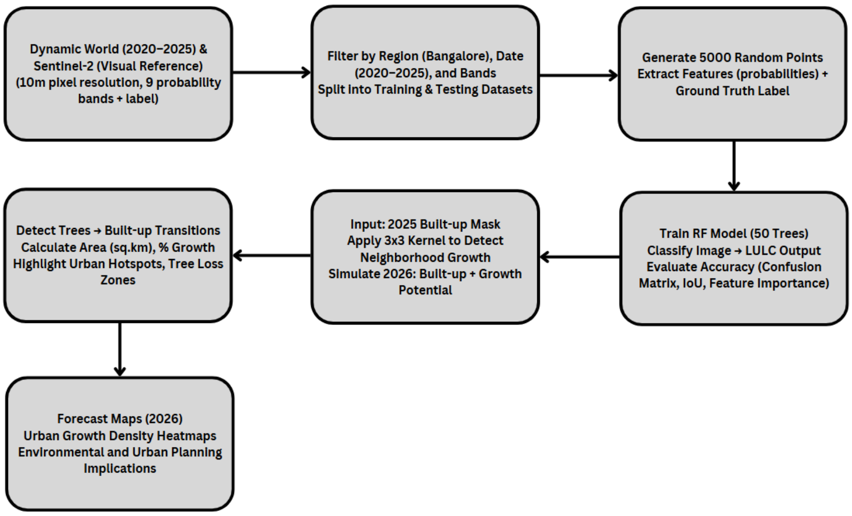


Fig. 2. Proposed Methodology for Urban Land Use Classification and Forecasting using Random Forest and Cellular Automata

5 Result Analysis

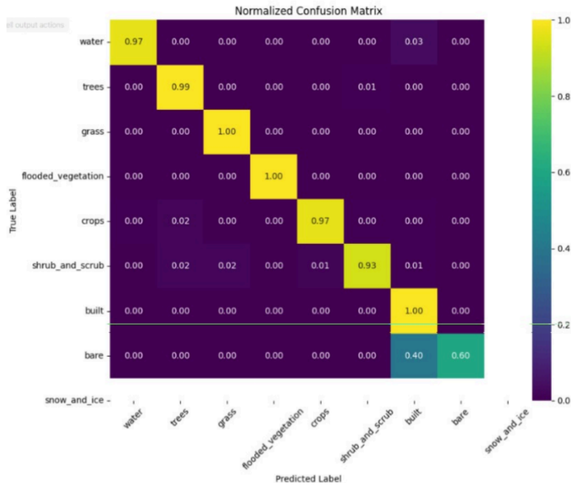


Fig. 3. Random Forest Confusion Matrix

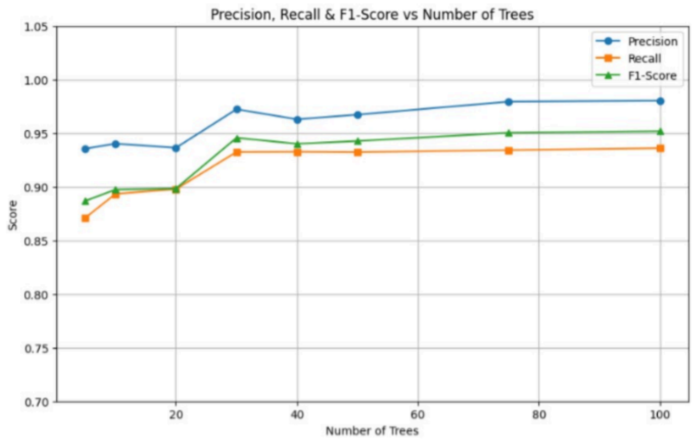


Fig. 4. Precision, Recall & F1 Score vs Number of Trees

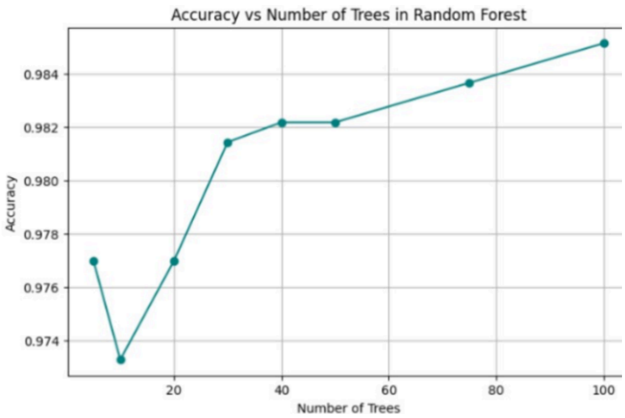


Fig. 5. Accuracy vs Number of Trees

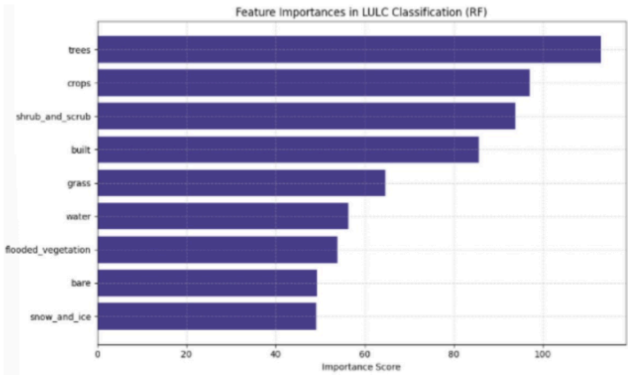


Fig. 6. Feature Importances in Random Forest Model

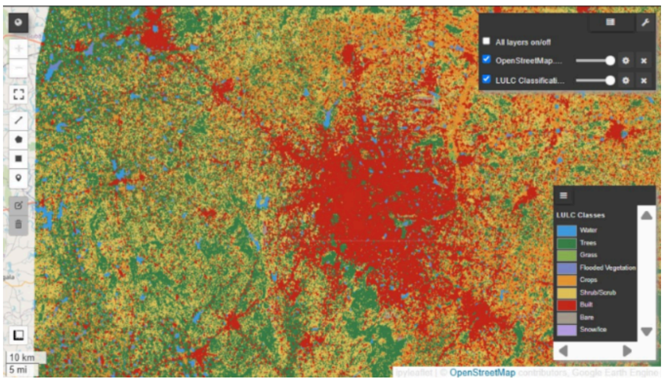


Fig. 7. LULC Classification Map using RF Model

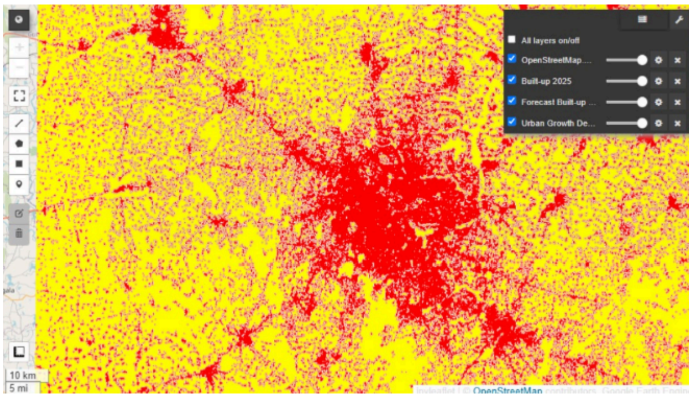


Fig. 8. Forecasted Built-up Area for 2026

Fig. 9. Projected Trees to Built-Up Transition (2025–2026)

Table 1. CA Evaluation Metrics.

Year Range	IoU Score	Tree → Built-up Area (sq.km)
2020-2021	0.8689	16.81
2023-2024	0.8638	29.77
2024-2025	0.8622	41.8

Table 2. Random Forest Evaluation Metrics.

Metric	Value	Notes
Overall Accuracy	0.9821	50 Trees
Kappa Coefficient	0.9752	Strong Agreement
Mean IoU	0.9004	High Class Balance

The proposed system demonstrated strong capability in classifying and forecasting land use and land cover (LULC) patterns over Bangalore using the Dynamic World dataset in conjunction with Sentinel-2 imagery. The Random Forest (RF) model, trained on 5000 sample points derived from band-wise class probabilities, effectively categorized land cover into nine distinct classes. These included vegetation types, built-up zones, and natural landforms. The classifier achieved a high overall accuracy of 98.21% and a Kappa coefficient of 0.975. These are indicative of strong agreement between predicted and actual labels (see Figure 3. RF Confusion Matrix).

Each class's performance was evaluated using Intersection over Union (IoU), with built-up areas reaching an IoU of 0.9855, and trees achieving 0.9765. The mean IoU across all classes stood at 0.9004, suggesting balanced model performance across diverse landscapes (see Table 2. Evaluation Metrics for Random Forest Classification).

To assess model robustness, metrics such as precision, recall, and F1-score were calculated for varying numbers of trees in the Random Forest. The analysis confirmed that 50 trees provided the best balance between accuracy and computational efficiency. The trends are plotted in Figure 4. Precision, Recall & F1 Score vs Number of Trees, with accuracy trends illustrated separately in Figure 5. Accuracy vs Number of Trees.

Feature importance analysis revealed that the ‘trees’ class was the most influential, followed by ‘crops’ and ‘shrub_and_scrub’, as shown in Figure 6. Feature Importances in Random Forest Model. This highlights the significant role of vegetative land in informing urban classification decisions.

The classification output for the test region is visualized in Figure 7. LULC Classification Map using RF Model, confirming a coherent spatial distribution of land use types in the region.

For forecasting, a Cellular Automata (CA)-based growth model was employed. The 2025 built-up area was extracted using the mode of Dynamic World labels, and the CA simulation projected likely expansion for 2026 using a 3x3 kernel neighborhood. The forecasted built-up area for 2026 was estimated at 1574.93 sq.km, compared to 1482.86 sq.km in 2025—marking a net increase of 92.07 sq.km and a growth rate of 6.21% (see Figure 8. Forecasted Built-up Area for 2026 and Table 1. Evaluation Metrics for CA-Based Urban Growth Forecasting).

To evaluate CA reliability, backcasting analysis was conducted for 2020→2021, 2023→2024, and 2024→2025. IoU scores of 0.8689, 0.8638, and 0.8622, respectively, validated the temporal coherence of the simulated growth (see Table 1).

One of the critical environmental insights of this study involves tree loss estimation. A focused simulation assessed tree-covered pixels in 2025 expected to convert to urban zones in 2026, resulting in a projected loss of 41.80 sq.km (see Figure 9. Projected Trees to Built-Up Transition (2025–2026)).

These results align closely with actual trends. For instance, between 2023 and 2024, 41.45 sq.km of tree land was observed to have been converted to built-up areas, with 16.81 sq.km between 2020 and 2021. CA simulation for 2023→2024 predicted 29.77 sq.km of tree loss, reaffirming the reliability of our approach (see Figure 10). Simulated Trees to Built-Up Transition (2023–2024)).

The study underlines the dual utility of this system: a robust LULC classifier powered by a well-tuned Random Forest, and a lightweight yet interpretable urban forecasting tool using Cellular Automata. While the CA model does not account for socio-political constraints or economic drivers, its ability to spatially simulate plausible urban growth offers a valuable decision-support resource for planners and sustainability advocates.

6 Conclusion

This study highlights the synergistic use of Random Forest (RF) and Cellular Automata (CA) for land use classification and urban expansion forecasting over Bangalore. The CA-RF framework effectively identified diverse land cover classes while forecasting plausible future urban spread with spatial consistency. Model performance was evaluated using accuracy, Kappa coefficient, IoU scores, and confusion matrix analysis, confirming high reliability across all classes. By simulating future growth and detecting ecologically sensitive transitions such as tree-to-built conversions, the system lays groundwork for proactive urban planning. Its ability to visualize and quantify change makes it a valuable tool for municipal authorities, sustainability analysts, and smart city initiatives aiming to balance urban development with ecological preservation.

References

1. Kamusoko, C., Gamba, J.: Simulating Urban Growth Using a Random Forest-Cellular Automata (RF-CA) Model. In: International Journal of Geo-Information, vol. 4(2), pp. 447–470. MDPI, Basel (2015).
2. Brown, C.F., Tuholske, C., Tellman, B., Evans, T.P., Wentz, E.A., Stokes, E.: Dynamic World, Near Real-Time Global 10 m Land Use Land Cover Mapping. In: Scientific Data, vol. 9, article 687. Nature Portfolio (2022).
3. Thottolil, J., Neelakantan, A., Anjana, N.M., Ganapathy, S.: Forecasting Urban Expansion of Bangalore Using CA-PLUS Model. In: Preprints.org (2023).
4. Raj, S., Gowtham, B.G.: Modelling and Prediction of Urban Growth Using SLEUTH Model – A Case Study of Mysuru City, India. In: IOSR Journal of Humanities and Social Science (IOSR-JHSS), vol. 29(1), Ser. 5, pp. 25–35. IOSR Journals (2024).
5. Yao, Y., Liu, X., Zhang, D., Liang, Z., Zhang, Y.: Simulation of Urban Expansion and Farmland Loss in China by Integrating Cellular Automata and Random Forest. In: arXiv preprint arXiv:1705.05651 (2017).
6. Zhang, Y., Li, X., Liu, X., Chen, Y.: Forecasting Urban Land Use Change Based on Cellular Automata and the PLUS Model. In: Land, vol. 11(5), article 652. MDPI, Basel (2022).
7. Girma, R., Fürst, C., Moges, A.: Land Use Land Cover Change Modeling by Integrating Artificial Neural Network with Cellular Automata-Markov Chain Model in Gidabo River Basin, Main Ethiopian Rift. In: Environmental Challenges, vol. 6, article 100419. Elsevier (2022).
8. Gidey, E., Dikinya, O., Sebego, R., Segosebe, E., Zenebe, A.: Cellular Automata and Markov Chain (CA_Markov) Model-Based Predictions of Future Land Use and Land Cover Scenarios (2015–2033) in Raya, Northern Ethiopia. In: Modeling Earth Systems and Environment, vol. 3, pp. 1245–1262. Springer (2017).
9. Fathizad, H., Rostami, N., Faramarzi, M.: Detection and Prediction of Land Cover Changes Using Markov Chain Model in Semi-Arid Rangeland in Western Iran. In: Environmental Monitoring and Assessment, vol. 187, article 629. Springer (2015).
10. Gharaibeh, A., Shaamala, A., Obeidat, R., Al-Kofahi, S.: Improving Land-Use Change Modeling by Integrating ANN with Cellular Automata-Markov Chain Model. In: Heliyon, vol. 6(5), article e05092. Elsevier (2020).
11. Habibie, M.I.: The Application of Machine Learning Using Google Earth Engine for Remote Sensing Analysis. In: Jurnal Teknologi Informasi, vol. 16(2), pp. 233–240. Universitas Muhammadiyah Surakarta (2022).
12. Gashaw, T., Bantider, A., Mahari, A.: Evaluations of Land Use/Land Cover Changes and Land Degradation in Dera District, Ethiopia: GIS and Remote Sensing Based Analysis. In: International Journal of Scientific Research in Environmental Sciences, vol. 2(5), pp. 199–208. International Scholars Journals (2014).
13. Hassan, Z., Shabbir, R., Ahmad, S.S., Malik, A.H., Aziz, N., Butt, A., Erum, S.: Dynamics of Land Use and Land Cover Change (LULCC) Using Geospatial Techniques: A Case Study of Islamabad, Pakistan. In: SpringerPlus, vol. 5, article 812. Springer (2016).
14. Gashaw, T., Tulu, T., Argaw, M., Worqlul, A.W.: Modeling the Impacts of Land Use/Land Cover Changes on the Hydrology of the Andassa Watershed, Blue Nile Basin, Ethiopia. In: Science of the Total Environment, vol. 619–620, pp. 1394–1408. Elsevier (2018).

Paper111

by Pradeep Kumar K

Urban Growth Forecasting and LULC Dynamics in Bangalore using Random Forest Classification and Cellular Automata on Dynamic World Satellite Data

Shruti Nigam[1], Rishitha Kattipallem[2], Bhuvana Mohini T N[3], Rangaraj BS[4]

[1] Dayananda Sagar University, Bangalore, Karnataka, India
shrutin567@gmail.com
eng21am0120@dsu.edu.in
[2] Dayananda Sagar University, Bangalore, Karnataka, India
Kattipallemrishitha315@gmail.com
eng21am0095@dsu.edu.in
[3] Dayananda Sagar University, Bangalore, Karnataka, India
bhuvana.m-aiml@dsu.edu.in
bhuvanamohinitn@gmail.com
[4] Dayananda Sagar University, Bangalore, Karnataka, India
rangaraj-cse@dsu.edu.in

Abstract. Urbanization represents one of the most impactful land use and land cover (LULC) changes in rapidly expanding cities, especially in developing countries. Bangalore, India, has experienced substantial urban sprawl in recent years, often replacing vegetated areas. Forecasting built-up growth accurately is vital for sustainable planning and environmental management. Existing models often depend solely on historical spatial patterns, lacking integration of spatial dynamics with machine learning. This study proposes a hybrid framework that merges Random Forest (RF)-based classification with Cellular Automata (CA) for spatial growth simulation. Using Dynamic World (DW) Sentinel-2 imagery from 2020 to 2025, the RF classifier is trained on class probability bands and evaluated using confusion matrix, Kappa, and mean IoU. Built-up expansion for 2026 is simulated using CA-based neighborhood logic on 2025 built-up clusters. The system also detects tree-to-built-up conversions, offering environmental insights. The implementation uses Google Earth Engine and Python libraries like geemap and seaborn. Achieving over 98% accuracy, the model predicts a 6.2% urban growth by 2026, supporting data-driven urban planning and land use monitoring.

Keywords: Land Use and Land Cover (LULC), Random Forest Classification, Cellular Automata Urban Forecasting.

1

Land Use and Land Cover (LULC) classification and forecasting are essential for monitoring urbanization, ecological change, and resource planning. In rapidly expanding cities like Bangalore, understanding land transitions—especially the growth of built-up areas—is vital for sustainable development and environmental management.

This study addresses two challenges: (1) accurate classification of current land cover using satellite imagery, and (2) forecasting future urban expansion through spatial modeling. While machine learning has improved LULC classification, predicting short-term land cover change remains difficult in heterogeneous urban areas, where vegetation, agriculture, and built-up zones shift rapidly. There is a pressing need for systems that not only classify land cover precisely but also model future changes with spatial context.

Open-access datasets like Sentinel-2 and Dynamic World, along with cloud platforms like Google Earth Engine (GEE), now support scalable analysis of spatiotemporal imagery. Random Forest classifiers have proven effective for LULC classification, while Cellular Automata (CA) offer spatially intuitive modeling of land transformation based on neighborhood dynamics.

Despite these advances, forecasting urban growth faces issues like intra-class variation and subtle transition zones. These challenges call for models that blend data-driven learning with interpretable spatial logic.

To address this, our work combines a Random Forest classifier (trained on Dynamic World class probabilities) with CA-based simulation to project built-up expansion for 2026, using patterns from 2020–2025. Implemented entirely on GEE, this approach also highlights environmental concerns by measuring green-to-urban transitions, particularly tree cover loss.

Our goal is to deliver a reproducible, interpretable, and scalable forecasting framework that supports informed urban planning, ecological monitoring, and sustainable development. The upcoming sections review related work, describe our methodology, present findings, and explore the real-world impact of our approach.

2

Urban growth and land use/land cover (LULC) classification have seen a range of approaches integrating machine learning and spatial dynamics. This section summarizes relevant works and their connection to our study.

[1] A hybrid RF-CA model was used to simulate urban expansion in Harare, Zimbabwe, where Random Forest generated transition potential maps that fed into a Cellular Automata model. The approach outperformed SVM-CA and LR-CA baselines, achieving a Kappa simulation accuracy of 0.51, highlighting the advantage of combining statistical and spatial modeling.

[2] The Dynamic World dataset, developed by Google and WRI, provides 10-meter resolution global LULC data using deep learning on Sentinel-2 imagery. It offers class probabilities across nine categories, making it well-suited for real-time urban monitoring and change detection tasks.

[3] In Bangalore, the CA-PLUS model was used to project urban growth up to 2030, integrating LULC data and socio-economic drivers. The study underlined the value of combining environmental and anthropogenic factors for enhanced urban forecasting accuracy.

[4] A study in Mysuru applied the SLEUTH CA-based model using Landsat data (1995–2022) and achieved 86.44% accuracy ($R^2 = 0.98$). This confirmed the efficacy of CA-based models in capturing temporal urban dynamics for regional planning.

Building on these efforts, our work applies the RF-CA framework with Dynamic World data to simulate urban growth in Bangalore for 2026. The integration of high-resolution, real-time LULC inputs with data-driven transition modeling addresses gaps in both accuracy and update frequency, especially critical for fast-developing regions.

3

3.1

The goal of this study is to develop an intelligent and interpretable land use and land cover (LULC) classification and forecasting system for the Bangalore region. The system aims to identify different land cover types using high-resolution satellite data and accurately forecast urban expansion patterns—specifically built-up area growth—by 2026.

To achieve this, the system integrates: A supervised machine learning model (Random Forest) for pixel-wise classification of Dynamic World Sentinel-2 data, A Cellular Automata (CA)-based neighborhood expansion approach for simulating future urban growth based on spatial patterns observed from 2020 to 2025.

The solution should ensure: Effective classification of land cover categories (e.g., trees, water, built-up, etc.) using multi-band input features, Forecasting of built-up area growth using spatial transition rules that mimic real-world urban sprawl tendencies, Quantitative assessment of model performance through accuracy, confusion matrix, Kappa score, Intersection over Union (IoU), and backcasting validations, Environmental insight by quantifying tree-to-built-up conversions and spatial density of predicted growth.

The system uses the Google Earth Engine (GEE) cloud platform and Dynamic World V1 dataset for near real-time land cover classification. The primary objective is to build a lightweight, explainable, and scalable forecasting model suitable for aiding city planners and researchers concerned with sustainable urban expansion.

3.2

Figure 1 illustrates the complete architecture of the proposed LULC classification and urban growth forecasting system. The system is built in two main stages: LULC Classification Module, Urban Growth Forecasting Module (Cellular Automata-Based).

Each stage works on top of satellite imagery and geospatial data from Google Earth Engine (GEE), using machine learning and spatial modeling to produce accurate land cover insights and simulate urban expansion.

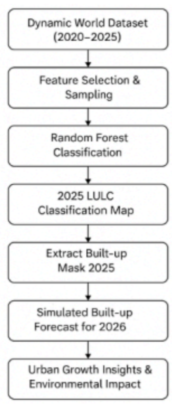


Fig. 1. System Architecture of the Proposed Model

The two-stage process begins with LULC (Land Use Land Cover) classification, where Sentinel-2 based Dynamic World images (2020–2025) for the Bangalore region are used as input. Probability bands for all land cover classes and the top-1 label band are extracted as features. Labeled training and testing points are generated through random sampling, and a Random Forest classifier is trained to classify each pixel into one of the nine Dynamic World land cover classes. Model performance is evaluated using metrics such as accuracy, Kappa coefficient, confusion matrix, and IoU, resulting in a classified LULC map for early 2025. In the second stage, urban growth forecasting is performed using a built-up mask derived from the 2020–2025 LULC classification. A Cellular Automata (CA)-based neighborhood expansion rule is applied, where pixels adjacent to existing built-up areas are marked as potential growth zones. This leads to the generation of a projected built-up map for 2026, followed by visualization and quantification of changes such as built-up area increase, tree-to-built conversions, and growth density heatmaps.

3.3

To model and forecast urban expansion using machine learning and spatial dynamics, this study relies on high-resolution, temporally consistent datasets from Google Earth Engine (GEE), specifically the Dynamic World (DW) and Sentinel-2 Harmonized datasets. The DW dataset, developed by Google and the World Resources Institute, provides near real-time 10-meter resolution land use/land cover (LULC) classifications with probabilistic estimates across nine major categories (e.g., trees, built-up, crops, water). Its “label” band, which identifies the most likely class, serves as a key input for supervised classification and temporal analysis. The dataset is central to our workflow—used for feature extraction, Random Forest (RF) training, and change detection.

Complementing this, Sentinel-2 Harmonized imagery contributes spectral bands for visual validation, cloud masking, and additional quality checks. Though DW is derived from Sentinel-2, direct access to raw bands enriches classifier training and visualization accuracy. Data is filtered to the Bangalore region from 2020 to 2025, and random sampling via GEE’s sample() function generates labeled points. These are split 70:30 into training and testing sets for performance evaluation using accuracy metrics and confusion matrices.

Despite its strengths, DW’s per-pixel classification may introduce noise in heterogeneous zones, and its use of maximum probability labels can oversimplify mixed-use areas. However, the dataset’s temporal granularity and spatial consistency make it highly suitable for LULC modeling. Together, DW and Sentinel-2 form a robust data foundation for simulating urban growth and assessing land transition dynamics over time.

4

This study combines machine learning classification with Cellular Automata (CA)-based forecasting to model future land use and land cover (LULC) changes, focusing on urban expansion in Bangalore. The entire pipeline is implemented on Google Earth Engine (GEE) using Python and Geemap, ensuring spatial precision and scalability for urban sustainability research.

We utilize the Dynamic World (DW) dataset, which offers 10-meter resolution land cover data from January 2020 to April 2025, containing probabilistic estimates for nine LULC classes along with a “label” band indicating the most likely class. Sentinel-2 Harmonized imagery supports visual inspection and data quality checks. Data is spatially filtered to the Bangalore region, and relevant bands are selected.

Feature vectors are generated using the DW class probabilities, with the “label” band as ground truth. A random sampling approach is used to extract 5000 points, which are split into 70% training and 30% testing subsets. A Random Forest (RF) classifier with 50 trees is trained to predict LULC classes. Classification performance is evaluated using accuracy, confusion matrix, precision, recall, F1-score, and Intersection over Union (IoU). The model achieves over 98% accuracy, with high IoU values, particularly for built-up and trees. Feature importance analysis reveals that “trees,” “crops,” and “shrub_and_scrub” are most predictive.

Forecasting is performed using a simple yet effective CA model. A 3x3 kernel is used to identify potential growth zones adjacent to existing built-up areas using focal_max(). The 2026 map is generated by merging current built-up zones with these likely expansion pixels. This neighborhood-based logic simulates short-term urban growth where planning data is unavailable.

The complete pipeline—data acquisition, classification, and forecasting—is illustrated in Figure 2. The system also detects meaningful transitions such as "trees to built-up," offering valuable inputs for sustainability planning. Metrics like projected built-up growth and vegetation loss align with Sustainable Development Goals (SDGs), highlighting the model’s broader utility in urban policy and environmental monitoring.

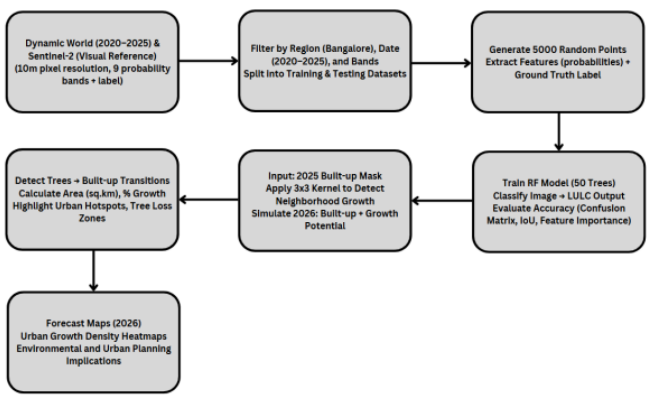


Fig. 2. Proposed Methodology for Urban Land Use Classification and Forecasting using Random Forest and Cellular Automata

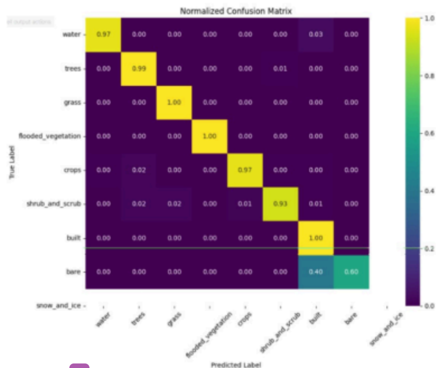


Fig. 3. Random Forest Confusion Matrix

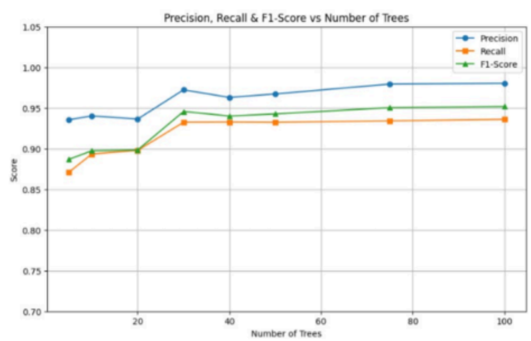


Fig. 4. Precision, Recall & F1 Score vs Number of Trees

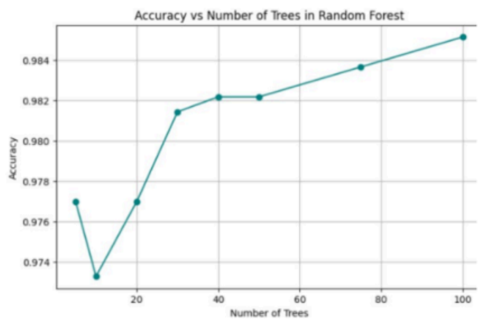


Fig. 5. Accuracy vs Number of Trees

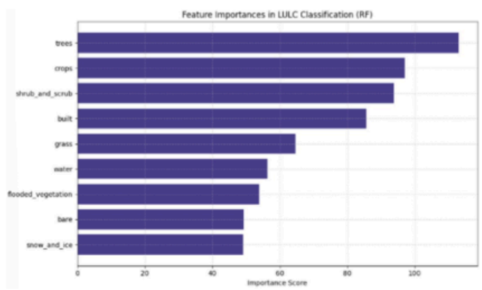


Fig. 6. Feature Importances in Random Forest Model

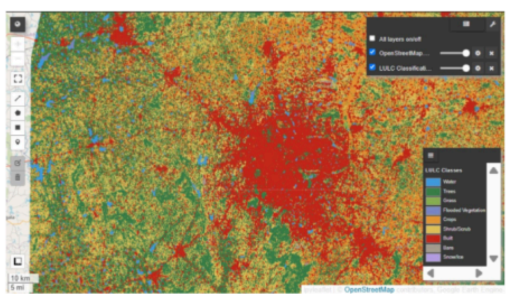


Fig. 7. LULC Classification Map using RF Model

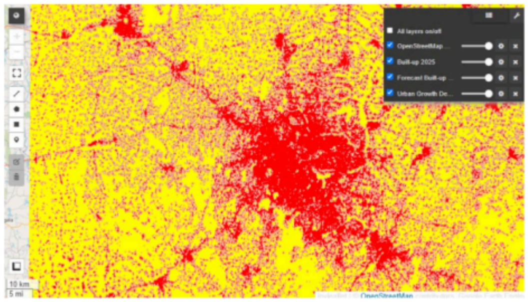


Fig. 8. Forecasted Built-up Area for 2026

Fig. 9. Projected Trees to Built-Up Transition (2025–2026)

Year Range	IoU Score	Tree → Built-up Area (sq.km)
2020-2021	0.8689	16.81
2023-2024	0.8638	29.77
2024-2025	0.8622	41.8

Table 1. CA Evaluation Metrics

Metric	Value	Notes
Overall Accuracy	0.9821	50 Trees
Kappa Coefficient	0.9752	Strong Agreement
Mean IoU	0.9004	High Class Balance

Table 2. Random Forest Evaluation Metrics

The proposed system demonstrated strong capability in classifying and forecasting land use and land cover (LULC) patterns over Bangalore using the Dynamic World dataset in conjunction with Sentinel-2 imagery. The Random Forest (RF) model, trained on 5000 sample points derived from band-wise class probabilities, effectively categorized land cover into nine distinct classes. These included vegetation types, built-up zones, and natural landforms. The classifier achieved a high overall accuracy of 98.21% and a Kappa coefficient of 0.975. These are indicative of strong agreement between predicted and actual labels (see Figure 3. RF Confusion Matrix).

Each class's performance was evaluated using Intersection over Union (IoU), with built-up areas reaching an IoU of 0.9855, and trees achieving 0.9765. The mean IoU across all classes stood at 0.9004, suggesting balanced model performance across diverse landscapes (see Table 2. Evaluation Metrics for Random Forest Classification).

To assess model robustness, metrics such as precision, recall, and F1-score were calculated for varying numbers of trees in the Random Forest. The analysis confirmed that 50 trees provided the best balance between accuracy and computational efficiency. The trends are plotted in Figure 4. Precision, Recall & F1 Score vs Number of Trees, with accuracy trends illustrated separately in Figure 5. Accuracy vs Number of Trees.

Feature importance analysis revealed that the 'trees' class was the most influential, followed by 'crops' and 'shrub_and_scrub', as shown in Figure 6. Feature Importances in Random Forest Model. This highlights the significant role of vegetative land in informing urban classification decisions.

The classification output for the test region is visualized in Figure 7. LULC Classification Map using RF Model, confirming a coherent spatial distribution of land use types in the region.

For forecasting, a Cellular Automata (CA)-based growth model was employed. The 2025 built-up area was extracted using the mode of Dynamic World labels, and the CA simulation projected likely expansion for 2026 using a 3x3 kernel neighborhood. The forecasted built-up area for 2026 was estimated at 1574.93 sq.km, compared to 1482.86 sq.km in 2025—marking a net increase of 92.07 sq.km and a growth rate of 6.21% (see Figure 8. Forecasted Built-up Area for 2026 and Table 1. Evaluation Metrics for CA-Based Urban Growth Forecasting).

To evaluate CA reliability, backcasting analysis was conducted for 2020→2021, 2023→2024, and 2024→2025. IoU scores of 0.8689, 0.8638, and 0.8622, respectively, validated the temporal coherence of the simulated growth (see Table 1).

One of the critical environmental insights of this study involves tree loss estimation. A focused simulation assessed tree-covered pixels in 2025 expected to convert to urban zones in 2026, resulting in a projected loss of 41.80 sq.km (see Figure 9. Projected Trees to Built-Up Transition (2025–2026)).

These results align closely with actual trends. For instance, between 2023 and 2024, 41.45 sq.km of tree land was observed to have been converted to built-up areas, with 16.81 sq.km between 2020 and 2021. CA simulation for 2023→2024 predicted 29.77 sq.km of tree loss, reaffirming the reliability of our approach (see Figure 10). Simulated Trees to Built-Up Transition (2023–2024)).

The study underlines the dual utility of this system: a robust LULC classifier powered by a well-tuned Random Forest, and a lightweight yet interpretable urban forecasting tool using Cellular Automata. While the CA model does not account for socio-political constraints or economic drivers, its ability to spatially simulate plausible urban growth offers a valuable decision-support resource for planners and sustainability advocates.

6

This study highlights the synergistic use of Random Forest (RF) and Cellular Automata (CA) for land use classification and urban expansion forecasting over Bangalore. The CA-RF framework effectively identified diverse land cover classes while forecasting plausible future urban spread with spatial consistency. Model performance was evaluated using accuracy, Kappa coefficient, IoU scores, and confusion matrix analysis, confirming high reliability across all classes. By simulating future growth and detecting ecologically sensitive transitions such as tree-to-built conversions, the system lays groundwork for proactive urban planning. Its ability to visualize and quantify change makes it a valuable tool for municipal authorities, sustainability analysts, and smart city initiatives aiming to balance urban development with ecological preservation.

Paper111

ORIGINALITY REPORT

5%	2%	3%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	"International Conference on Intelligent Data Communication Technologies and Internet of Things (ICICI) 2018", Springer Science and Business Media LLC, 2019 Publication	1%
2	M Gowtham, M. K. Banga, M allanagouda Patil. "Secure Internet-of- Things: Assessing Challenges and Scopes for NextGen Communication", 2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), 2019 Publication	1%
3	www.researchgate.net Internet Source	1%
4	Funani, Tshivhase. "Monitoring and Evaluating Urban Land Use Land Cover Change Using Machine Learning Classification Techniques: A Case Study of Polokwane Municipality.", University of the Witwatersrand, Johannesburg (South Africa) Publication	1%
	essay.utwente.nl	

5	Internet Source	<1 %
6	Meng Yuan, Lan Li, Hangnan Yu, Jiapeng Xiong, Jiangtao Yu. "Considering human interference to prioritize spatial conservation in a transboundary river basin using Zonation", Springer Science and Business Media LLC, 2025 Publication	<1 %
7	github.com Internet Source	<1 %
8	www.researchsquare.com Internet Source	<1 %
9	d197for5662m48.cloudfront.net Internet Source	<1 %
10	www.mdpi.com Internet Source	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		