A

Major Project Presentation

on

helioHarvest: Automated Building Footprint Extraction and Rooftop Solar Potential Estimation

in

Fulfillment for Final Year Computer Engineering Course- BTech Computer Engineering(Regional Language)

Aadit Kisanrao Palande	**121B1D011**
Aditya Atul Kode	**121B1D014**
Devashish Sanjay Gaikwad	**121B1D017**
Prasad Padmakar Joshi	**121B1D018**

(A. Y. 2024-25)

Under the guidance of

Dr. Rachana Y. Patil

Department of Computer Engineering

PCET's Pimpri Chinchwad College of Engineering

Contents

Topic	Slide no.
Introduction	3
Motivation	4
Objectives	5
Software requirements	6
Datasets	7
Algorithms	8
Software Testing	9
Literature Review	10
Proposed Method	14
Conclusion	18
Paper Publication Details	19
References	20

Introduction

- This project introduces helioHarvest, a web-based system designed to estimate rooftop solar energy potential using geospatial tools and machine learning.

- It addresses the high cost and complexity of traditional assessments, offering an accurate, automated, and accessible alternative.

- It achieves this by allowing users to select rooftop areas through an interactive map interface, fetching real-time and historical solar irradiance data via APIs, and applying advanced machine learning models like Random Forest and XGBoost to generate precise energy output predictions.

- The system provides intuitive visual analytics for energy yield, cost savings, and environmental impact.

Motivation

- **Energy Independence** – Enabling households and regions to rely less on imported fuels.
- **Technological Innovation** – Advancing the use of geospatial and ML tools in energy systems.
- **Precision and Optimization through Technology** – Traditional solar assessments are costly and slow; helioHarvest uses Mapbox GL, PVGIS, and ML to deliver precise, automated predictions, making solar energy practical for homeowners and planners.
- **Cost-Effectiveness and Accessibility** – High-cost conventional methods limit solar adoption; helioHarvest leverages affordable web tech (React.js, Node.js) and open data to democratize rooftop solar insights across India and beyond.
- **Scalable Urban Solutions** – Supporting city-wide solar adoption for sustainable development.
- **Economic Benefits** – Cutting electricity costs and boosting local renewable energy markets.
- **Environmental Awareness** – Encouraging eco-conscious choices through data-driven insights.

Objectives

- **To design and develop** a web-based product (*helioHarvest*) that automates rooftop solar energy potential estimation, addressing the limitations of manual surveys.

- **To solve the challenge** of inaccessibility, high cost, and inaccuracy in traditional solar assessments, especially in developing and remote regions.

- **To build an end-to-end solution** that integrates geospatial mapping, real-time weather data, and machine learning for accurate, scalable, and efficient predictions.

- **To empower users**—homeowners, planners, and policymakers—with data-driven insights that support clean energy adoption and sustainability goals.

- **To create a user-centric platform** that provides visual analytics for energy yield, cost savings, and environmental benefits, making solar decisions easier and smarter.

Software Requirements

- **Goal:** Cross-platform, scalable, performant solar estimation.
- **Operating System:**
  - Dev: Win/macOS.
  - Deploy: Browsers; mobile-ready.
- **Languages:**
  - JS: Frontend (React), Backend (Node).
  - Python: ML (Scikit-Learn, Pandas).
  - HTML/CSS: Frontend (Tailwind).
- **DB:** API: PVGIS, Tomorrow.io.
- **Tools:**
  - VS Code: IDE.
  - Node.js: Backend (Express, Axios).
  - Jupyter: ML.
  - Git/GitHub: Version control.

Datasets

- **Source**: National Solar Radiation Database (NSRDB)
  - Provides reliable, satellite-derived solar resource data, crucial for accurate solar energy assessments.
- **Processing:** Data aggregated to monthly averages
  - Transforms high-resolution data into a suitable format for modeling monthly solar energy patterns. This simplifies the analysis and aligns with the project's goals.
- **Scope:** Specific cities in India: Mumbai, Navi Mumbai, Pune, Nashik, Nagpur
  - Focuses the analysis on a geographically relevant region, allowing for tailored insights into solar energy potential in these urban areas.
- **Features:**
  - **Month:** Captures seasonal variations in solar radiation, a fundamental driver of solar energy availability.
  - **Solar Zenith Angle_Month_Avg:** Represents the angle of the sun, a primary factor determining the intensity of solar radiation received at a location.
  - **Relative Humidity_Month_Avg:** Accounts for the impact of atmospheric moisture on solar radiation transmission, as humidity can affect scattering and absorption.
  - **Wind Speed_Month_Avg:** Indirectly influences solar energy production by affecting panel temperature and potentially cloud cover patterns.
- **Target (Output) Variable:**
  - **solar_energy_per_unit_area (kWh/m^2):** The quantity we aim to predict, representing the solar energy available per unit area, a key metric for solar energy system design and analysis.

Algorithms

- **Models Implemented:** Stacking Regression Model, Linear Regression, Decision Tree Regressor, Random Forest Regressor, Support Vector Regression (SVR), Gradient Boosting Regressor, XGBoost Regressor.

- **Algorithm Descriptions:**
  - Linear Regression: Models linear relationships between variables.
  - Decision Tree: Partitions data into regions for prediction.
  - Random Forest: Ensemble of multiple decision trees.
  - SVR: Regression using support vectors and kernels.
  - Gradient Boosting: Sequential boosting of weak learners (trees).
  - XGBoost: Optimized gradient boosting implementation.
  - Stacking: Combines predictions from base models.

- **Rationale for Model Selection:**
  - Diverse set of algorithms
  - Ensemble methods (Random Forest, Gradient Boosting, XGBoost, Stacking)
  - Stacking Regressor

Software Testing

1. **Unit Testing:**
   - Objective: Validate individual components to ensure they function correctly in isolation.
     - Turf.js Area Calculation
     - Solar Potential Formula
     - Backend Express API (endpoints)
     - Frontend Axios Requests (HTTP, errors)
1. **Integration Testing:**
   - Objective: Confirm smooth interaction between interconnected modules.
     - Turf.js + Axios + Express API (area → potential)
     - Mapbox + Marker Logic (geolocation, drawing)
     - API + Monthly Data (irradiance, loss factors)
1. **System Testing:**
   - Objective: Assess the overall system's performance under real-world scenarios.
     - End-to-End Flow (UI input → backend → UI output)
     - Real-time Interaction (multi-user, locations)
     - Performance under Load (large polygons, API calls)
     - Map/UI Rendering (devices, sizes)

Literature Review - (Summary)

Authors	Paper Focus	Methodology	Features	Advantages	Limitations
Villa-Ávila et al. (2023)	Methodology for rooftop PV generation in urban areas	Roof-Solar-Max, GIS, 3D models	Geospatial and 3D modeling for shading and placement	High spatial accuracy, promotes self-consumption	Limited scalability beyond urban study area
Rees et al. (2022)	Feasibility of rooftop solar in northern latitudes	Airborne LiDAR, OpenStreetMap	Geospatial solar analysis for extreme climates	Viability in low sunlight regions	Limited to Tromsø; high-res data needed
Andreevska et al. (2022)	Identification of PV-suitable rooftops	GIS, LiDAR, multicriteria analysis	Considers slope, aspect, and solar radiation	Scalable spatial analysis	Data-dependent, limited region-specific application
Singla et al. (2023)	Urban PV estimation in India	High-res satellite imagery, DEM	Urban shadowing and vegetation analysis	Fine-grained insights for Indian cities	Costly high-resolution data required
Cenký and Krajňák (2022)	Low-cost rooftop PV estimation	QGIS, shadow analysis	Open-source tools and public data	Accessible and replicable method	Lower accuracy from basic shadow modeling

Literature Review - (Summary)

Authors	Paper Focus	Methodology	Features	Advantages	Limitations
Cenký and Šúri (2022)	Large-scale solar capacity evaluation	GIS software, public datasets	Irradiation and PV output estimation	Scalable and cost-effective	Depends on quality of public data
de Luna and Javaid (2023)	Machine learning for PV prediction	LASSO Regression	Predictive modeling of rooftop capacity	High accuracy (R^2 = 0.9987)	Requires computational skills
Sander et al. (2022)	Germany-wide rooftop PV potential	Satellite & reanalysis data	5×5 km grid-based national analysis	Useful for policy and planning	Lower spatial resolution
Wang et al. (2023)	AI-enhanced rooftop PV assessment	Deep learning	Rooftop classification by orientation	Reduces misclassification, high accuracy	High data and computing demand
Rahman et al. (2022)	Grid independence through PV	GIS, 3D modeling	Optimizes spatial panel layout	Cost-efficient and independence-focused	Limited to specific urban context
Schinke et al. (2023)	PV and EU solar regulations	Not specified	Alignment with EU policy initiatives	Policy-aligned insights	Needs empirical validation

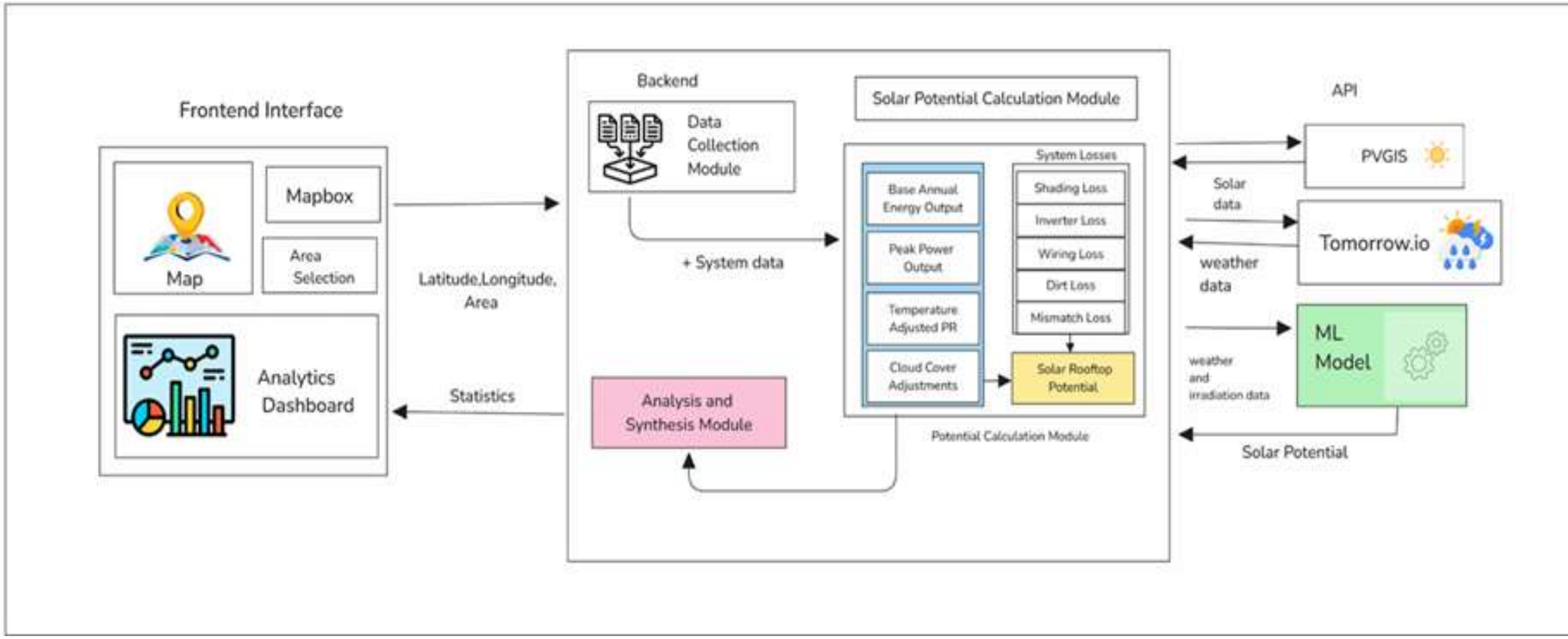
Literature Review - (Summary)

Authors	Paper Focus	Methodology	Features	Advantages	Limitations
Zhang et al. (2023)	Automated rooftop PV estimation	Deep learning, UAV	Drone-captured high-resolution data	Reduces manual effort	Requires UAV and processing tech
Torres et al. (2022)	Function-based solar estimation	Remote sensing	Assesses by building type (residential/commercial)	Better mixed-use estimates	Classification challenges
Müller et al. (2023)	Precision PV potential	High-res imagery, DSM	High-accuracy rooftop analysis	Reliable detail in output	Needs quality DSM input
Patel et al. (2022)	Fully automated PV estimation	UAV photogrammetry	Drone-based shading and angle detection	Efficient and reduced errors	Limited by UAV operations
Meier et al. (2022)	PV potential via ray tracing	Backward ray tracing	Detailed modeling of shading	Very high shading precision	Resource-intensive

Literature Review - (Summary)

Authors	Paper Focus	Methodology	Features	Advantages	Limitations
Choi et al. (2023)	City-wide solar analysis	Deep learning	District-level solar mapping	Scalable for urban plans	Needs DL infrastructure
Verma et al. (2022)	Rooftop extraction using AI	Machine learning	Identifies rooftops for PV	Speeds up data processing	Needs robust training data
Kim et al. (2023)	High-density urban PV estimation	Deep learning, GIS	Handles complex rooftops in cities	Effective in limited open spaces	High computational needs
Okafor et al. (2022)	PV in developing regions	GIS, geospatial analysis	Case study in Nigeria	Promotes adoption in underrepresented regions	Limited local data availability

Proposed Method



The proposed methodology for **helioHarvest** leverages an interactive web application to estimate rooftop solar energy potential. Users input rooftop coordinates via a Mapbox-powered map, after which real-time weather and historical solar data are fetched from external APIs (PVGIS and Tomorrow.io). A mathematical model, combined with machine learning algorithms (Random Forest, XGBoost, etc.), processes the data to provide accurate energy yield predictions, accounting for system losses, environmental factors, and seasonal variations. The results, including monthly and yearly energy output, cost savings, and environmental impact, are visualized in an intuitive dashboard for easy interpretation, offering users a comprehensive, data-driven solution for solar adoption.

User Interaction & Input

- Users open the web application and view a Mapbox-powered interactive map.
- They zoom to their building, then draw a polygon around the rooftop.
- The system captures the rooftop's latitude/longitude coordinates and computes its area via the shoelace formula.

Data Acquisition

- **Historical Irradiance**: The backend calls the PVGIS API to retrieve monthly average solar irradiation (G_{avg}) for the rooftop coordinates.
- **Real-Time Weather**: Simultaneously, the Tomorrow.io API provides current cloud cover, temperature, and other meteorological data.

Preprocessing & Loss Adjustment

- The raw irradiation and weather data are cleaned and aligned to the selected rooftop area.
- A month-specific system loss factor (L)—accounting for shading, soiling, inverter/wiring losses, and temperature derating—is applied.

Core Solar Potential Calculation

- For each month, the system computes daily energy yield:

  $E = (0.7 \times A) \times G_{av}g \times \eta \times (1 - L)$

  - A = usable rooftop area (m^2)
  - 0.7 = derating for wiring/inverter inefficiencies
  - η = panel efficiency (20%)
  - L = combined loss factor

- Monthly yield: $E_{month} = E \times days\_in\_month$
- Annual yield: $E_{\gamma ear} = \sum_{1}{}^{2}{}_{12}\ E_{month}$

Machine-Learning Enhancement

- Six regression models (Linear, Random Forest, SVR, Gradient Boosting, XGBoost, Decision Tree) are trained on historical and real-time data.
- A stacking ensemble combines their predictions to reduce error (MAE, RMSE) and adjust the calculated energy yield.

1. **Analytics & Visualization**
   - The Analysis & Synthesis module merges calculated and ML-refined estimates.
   - Results—including monthly/annual kWh, cost savings, and CO_2 reduction—are sent to the frontend.
   - The dashboard renders interactive charts, graphs, and summary statistics for user interpretation.
2. **API Orchestration**
   - All data exchanges occur via REST endpoints:
     - /proxyIrradiationData for PVGIS calls
     - /proxyWeatherData for Tomorrow.io calls
     - /calculatePotential for core math and ML inference
   - Robust error handling and caching ensure continuity during API outages.

This end-to-end pipeline—spanning user input, geospatial computation, data integration, mathematical modeling, and machine-learning refinement—delivers fast, accurate, and accessible rooftop solar potential estimates.

Conclusion

This research has presented a comprehensive framework for estimating rooftop solar potential by integrating geospatial analysis, machine learning, and real-time environmental data. The helioHarvest mathematical model provides a robust foundation for calculating energy generation and potential savings, while the modular architecture ensures flexibility and scalability. Experimental analysis confirms the significant impact of environmental factors on solar energy production, with seasonal variations demonstrating the importance of long-term data in accurate predictions. The correlation analysis between different parameters offers valuable insights for optimizing solar panel deployment and operation.

Paper Publication Details

- Title of Paper: **helioHarvest: Automated Building Footprint Extraction and Rooftop Solar Potential Estimation**
- Name of International Journal / conference : **10th Edition ICT4SD International ICT Summit & Awards**
- Status : **Submitted for review**
- Name of Authors: **Devashish Sanjay Gaikwad, Aadit Kisanrao Palande, Prasad Padmakar Joshi, Aditya Atul Kode**
- Date of paper submission: **15 April 2025**
- Indexing: **Scopus**

References

For example (Follow given format):

1. Gnoying Feng et.al “Experimental research on vertical axis wind turbine” IEEE school of energy and power engineering, vol. 978,no.1, 2009.

2. G.M.Hasan Shaharirar “Design and construction of vertical axis wind turbine” IEEE International form on strategic technology, vol. 978, no. 1, pp 326-329, Oct 2014.

3. S.Sathiyamoorthy et.al, “Hybrid energy harvesting using Piezoelectric materials, automatic rotational solar panel, vertical axis wind turbine”, International conference on modelling, optimisation and computing, IEEE, Vol. 38, no. 10, pp 843-852, July 2005.

4. Atif Shahzad et. Al. “ Performance of a Vertical axis wind turbine under accelerating and decelerating flows” Second international through-life engineering service conference, Elsevier, Vol. 11, no. 1, pp 311-316, 2013.

5. Sahishnu R shah et. Al., “ Design, modelling and economic performance of a vertical axis wind turbine,” Energy Reports, Elsevier, Vol 11, pp. 619-623, 2018.

Thank you