EcoInnovators Ideathon 2026 Challenge: AI-Powered Rooftop PV Detection

1. Challenge Overview

PM Surya Ghar: Muft Bijli Yojana is a government scheme that aims to provide free electricity to households in India. The scheme was launched by Prime Minister Narendra Modi on February 15, 2024. With an investment of over Rs. 75,000 crores, the scheme aims to light up 1 crore households by providing up to 300 units of free electricity every month.

Governance for this scheme include the need to verify rooftop solar installations, so that subsidies reach genuine beneficiaries and public trust remains high. Field inspections alone are slow, costly, and uneven across states and DISCOMs. This challenge asks you to build a governance-ready, auditable, and low-cost remote verification digital pipeline that answers a simple question at any given coordinate (latitude, longitude):

“Has a rooftop solar system actually been installed here?”

Your system should work reliably across India’s diverse roof types (sloped, flat roofs) and imaging conditions. The goal is not only accuracy, but also auditability and generalization across states.

2. Rooftop PV system

What is a Rooftop Solar System?

A rooftop solar PV system converts sunlight into electricity using solar panels mounted on building roofs. This electricity can be:

- Consumed within the building (self-consumption),
- Exported to the grid under net metering or gross metering,
- Stored in batteries (optional, depending on configuration).

Main Components of a Rooftop Solar System

A. Solar PV Modules

These are the main power-generating units.

Common types in India: Polycrystalline, Mono PERC, bifacial

Efficiency: Ratio of sunlight converted to electricity (typically 18–21%)

Modules are mounted facing south with optimal tilt to maximise annual energy generation.

B. Mounting Structure

The structure holds the panels in place.

- Made of aluminium or galvanised steel

- Designed to withstand wind loads (as per IS 875)

Good mounting ensures structural safety and optimum solar incidence.

C. Solar Inverter

Converts the DC power from modules into usable AC power.

D. Balance of System (BoS)

This includes everything except modules and inverters like cables, junction boxes etc.

E. Optional: Battery Energy Storage System (BESS)

Used primarily for backup power in outage-prone areas.

How Does a Rooftop Solar System Work? (Simplified Flow)

1. Sunlight hits the PV modules.

2. Panels generate DC electricity.

3. Inverter converts DC → AC.

4. Electricity first supplies home/office loads.

5. Excess energy is either:

   - Exported to the grid (net metering), or

   - Stored in batteries (if available).

6. A bidirectional meter records import and export.

3. Core Objectives (What your system must do)

When given an input folder location with a xlsx containing list of sample id and corresponding geographic coordinates (latitude and longitude) and output folder location, your code must

1. Fetch: Automatically retrieve or accept a recent, high-resolution rooftop image for a given (lat, lon).
2. Classify: Run inference using your trained model and determine whether rooftop PV is present (binary: present / not present) within a 1200 sq. ft radius buffer zone. If not present,

determine whether rooftop present within a 2400 sq. ft radius buffer zone. Return a calibrated confidence score (if applicable).

3. Quantify: If PV is present, estimate total area of panel (m^2) which has the largest overlap with the selected buffer zone.
4. Explainability: Produce audit-friendly artifacts — polygon mask or bounding boxes, logic to compute area and a quality control (QC) status.
5. Store: JSON file and artifacts into output folder location

Required QC status values:

- VERIFIABLE — Clear evidence either way (present/not present).
- NOT_VERIFIABLE — Insufficient evidence (e.g., low resolution, heavy shadow/cloud, occlusion by tanks/trees, stale imagery).

4. Inputs & Outputs

Input data (available in xlsx file):

- sample_id
- latitude, longitude (WGS84) — may include small geocoding jitters

Mandatory output per site (JSON record):

```
{
  "sample_id": 1234,
  "lat": 12.9716,
  "lon": 77.5946,
  "has_solar": true,
  "confidence": 0.92,
  "pv_area_sqm_est": 23.5,
  "buffer_radius_sqft": 1200,
  "qc_status": "VERIFIABLE",
  "bbox_or_mask": "<encoded polygon or bbox>",
  "image_metadata": {"source": "XYZ", "capture_date": "YYYY-MM-DD"}
}
```

Human-readable artifacts:

- Audit overlay PNG/JPEG with polygon or bounding boxes and legend (class, confidence, area).

5. Provided Data & Resources

You will receive:

- Training dataset (EI_train_data.xlsx): id, lat, long, has_solar

You may use additional permissible imagery sources if they respect licensing and privacy. For example, you may use Google static maps Places API more info. Clearly cite every external source you use.

6. Evaluation Criteria (Weighted)

Your final score is a weighted aggregate across accuracy, auditability, efficiency, and ethics/operability.

Criterion	Metric / Evidence	Weight (%)
Detection accuracy	F1 score on has_solar	40
Quantification quality	RMSE for PV area (m^2)	20
Generalization & robustness	Performance across diverse states/roof types; handling look-alikes	20
Others – code quality, documentation, usability		20

7. Deliverables (What to submit)

- GitHub repository: clean code, README, clear run instructions for API/CLI. more info
- Docker image: Name, repository and tag of the Docker image of your environment stored in Docker hub more info
- Trained model file (if applicable): .pt, .joblib, .pkl etc file of trained model
- Model card (2–3 pages): data used, assumptions, logic, known limitations/bias, failure modes, and retraining guidance. more info
- Prediction files: JSON/CSV and artifacts, following the schema above.
- Model Training Logs: A report or clear export of training logs (e.g., via CSV or MLflow export) documenting key metrics (Loss, F1 Score, RMSE) across training epochs/steps. more info

8. Rules & Eligibility

- You may use open-source libraries and permissible datasets with attribution.
- No use of private or illegally obtained imagery.
- Do not hard-code answers for the test set; submissions will be integrity-checked.
- Share a short license statement (preferably an OSI-approved license) in your repo.
- Use only permissible imagery and respect licensing; cite all sources clearly.
- Document known biases (e.g., urban vs rural performance gaps) and mitigation steps.