TerraStack Bio-Residue Estimation System

Scientific Methodology and Technical Architecture

TerraStack Engineering Team

January 26, 2026

Abstract

This report details the technical architecture and scientific methodology of the **TerraStack Bio-Residue Estimation System**. The system is a geospatial API designed to estimate agricultural crop residue availability at the farm-plot level. It integrates satellite-derived plot data with a multi-tier yield estimation engine. The system evaluates residue quantification methodologies and implements a robust fallback hierarchy (Crop Cutting Experiments → APY → National Averages) to address data scarcity. We document the final system structure, including the experimental ET-based yield module, and validate the approach using a case study from Yavatmal, Maharashtra.

Contents

1 Introduction

The **TerraStack Bio-Residue Estimation System** is a specialized geospatial intelligence module designed to bridge the critical data gap between agricultural production and biomass supply chains. As the biofuel and bio-energy sectors expand to meet global sustainability mandates, the ability to accurately quantify feedstock availability at a granular level has become a prerequisite for operational viability.

This system serves as a high-precision digital auditor, converting satellite-derived farm plot data into actionable inventory logs for biomass aggregators, pellet manufacturers, and sustainability consultants.

1.1 Types of Agricultural Residues

The system specifically targets **Above Ground Dry Mass** which is the non-edible plant material generated as a by-product of agricultural cultivation. The system quantifies residues of two primary categories:

- **Field-Based Residues:** Biomass left on the farm plot after mechanical or manual harvest.
  - *Examples:* Cotton stalks, Maize stover, Soybean straw, Wheat straw.
  - *Logistics:* Requires baling and collection directly from the field.
- **Process-Based Residues:** Biomass generated at a processing facility (e.g., mills or threshing centers).
  - *Examples:* Rice husks, Sugarcane bagasse, Groundnut shells.
  - *Logistics:* Generally available at centralized aggregation points.

1.2 Strategic Importance

In the absence of direct measurement, biomass procurement has historically relied on rough regional estimates (like multiplying total district area by a generic factor). This approach leads to severe supply chain inefficiencies:

1. **Feedstock Insecurity:** Plants are often located in areas with theoretical surplus but practical deficit.
2. **Logistical Blindness:** Aggregators lack visibility on *when* and *where* harvest is occurring in real time.
3. **Audit Failure:** Without plot-level traceability, biomass cannot easily qualify for high-value carbon credits.

1.3 System Capabilities & Output Ranges

The TerraStack system resolves these challenges by delivering plot-level precision. Unlike static statistical reports, the API generates dynamic, location-specific intelligence.

Range of Outputs:

- **Granularity:** Individual farm plots (< 0.5 ha) up to regional clusters.
- **Quantification:** Dry Matter (DM) tonnage estimates ranging from 0.1 tonnes (small plots) to $> 1000t$
- **Temporal Data:** Harvest window predictions to align procurement teams with harvester activity.

- **Confidence Metrics:** Every estimate is tagged with a confidence score ($0.0 - 1.0$) based on the quality of the underlying satellite classification.

2 Scientific Framework

2.1 Variable Definitions

The system utilizes the following standard variables for calculations:

- A **(ha):** Crop area inside the plot for a season.
- Y **(t/ha):** Economic yield (harvested product) per hectare.
- R **(t/ha):** Above-ground residue yield per hectare (straw/stalks/leaves).
- $R_{\textbf{plot}}$ **(t):** Plot-level residue mass, $R_{\text{plot}} = A \cdot R$.
- HI **(-):** Harvest Index, $HI = \frac{Y}{Y+R}$.
- RPR **(-):** Residue-to-Product Ratio, $RPR = \frac{R}{Y}$.
- ET_0 **(mm):** Reference evapotranspiration (climate water demand).
- K_c **(-):** Crop coefficient ($ET_c = K_c \cdot ET_0$).
- ET_a **(mm):** Actual crop evapotranspiration.
- K_y **(-):** Yield response factor (water-stress sensitivity).
- Y_m **(t/ha):** Maximum yield under non-stressed conditions.
- **Rainfall Anomaly:(%)** A metric indicating the deviation of actual rainfall from the long-term average (LTA) for a specific location and period. A positive value signifies surplus rainfall above the norm, implying lack of water stress.

2.2 Evaluated Methodologies

We evaluated three potential methods for converting Economic Yield (Y) to Residue (R) before selecting the final architecture:

- **Method A: Linear Regression (Selected Standard)**
- **Method B: Harvest Index (HI)**
- **Method C: Residue-to-Product Ratio (RPR)**

2.2.1 Method: Linear Model (IPCC-style)

$$R = \alpha + \beta Y$$

This model accounts for residue generated even at low yields via the intercept α. It is the most robust method for cereals and pulses and is the primary method(under testing) implemented in TerraStack.

- **Viability:** High. Recommended by IPCC 2019 Refinement.
- **Implementation:** Used for Cotton, Soybean, Wheat, Gram.

2.2.2 Aliter Method 1: Harvest Index (HI)

$$R = Y \left(\frac{1 - HI}{HI} \right)$$

Partitions total biomass based on biological efficiency.

- **Viability:** Moderate. Useful when specific α, β coefficients are missing but biological data exists.
- **Implementation:** Could have been used as a secondary fallback for specific crop varieties.

2.2.3 Aliter Method 2: Residue-to-Product Ratio (RPR)

$$R = Y \cdot RPR$$

A simple multiplier.

- **Viability:** Low for field crops (overestimates at low yields), but High for processing by-products.
- **Implementation:** Helpful specifically for Sugarcane Bagasse estimations. Does not scale well for field crops

3 Final System Architecture

Based on the evaluation above, the final system was architected as a Python-based REST API.

3.1 File Structure

File	Purpose
`app/utils/residue_utils.py`	**Core Logic**: Contains the Fallback Hierarchy and IPCC coefficients.
`app/api/endpoints/crop_yield.py`	**API Endpoint**: Handles REST requests (‘/estimate‘).
`app/utils/residue_grapher.py`	**Visualization**: Generates PNG dashboards.
`maharashtra` (DB)	**PostGIS Database**: Stores CCE data.

Table 1: System Component Overview

3.2 Data Sources

The system aggregates data from four distinct sources to ensure coverage:

1. **CCE Database:** Ground Crop Cutting Experiments (highest accuracy).
2. **District APY:** Historical district-level statistics (2019).
3. **State APY:** State-level aggregated yields (2021-2026).
4. **CROP_COEFFICIENTS:** India-wide averages (fallback).

4 Algorithm: Yield Fallback Hierarchy

The system employs a strict decision tree to resolve Economic Yield (Y).

Algorithm 1 Yield Resolution Strategy

```
Require: Latitude, Longitude, Crop, District
Ensure: Yield (Y)
Check 1: True values (CCE)
if CCE data available within 50km radius then
    Y ← Query_PostGIS_CCE(District, Crop)
    return Y
end if
Check 2: District Stats (APY)
if District CSV has entry for (District, Crop) then
    Y ← Lookup_District_APY(District, Crop)
    return Y
end if
Check 3: State Stats
if State CSV has entry for (State, Crop) then
    Y ← Lookup_State_APY(State, Crop)
    return Y
end if
Check 4: National Average
Y ← CROP_COEFFICIENTS[Crop]["india_avg"]
return Y
```

Table 2: **Yield Data Source Example:** Amravati, Maharashtra (20.93°N, 77.47°E)

Crop	CCE	District APY	India Avg	Selected Source	Yield (kg/ha)
COTTON	✓	✗	✓	CCE (Ground Truth)	**1,005**
GRAM	✗	✗	✓	National Avg (Fallback)	**1,200**
SORGHUM	✗	✓	✓	District APY	**929**

5 Experimental Module: ET-Based Estimation

An alternative estimation method is included in the codebase for **testing** purposes. This implements the FAO Water Production Function to estimate yield based on hydrological stress.

5.1 Model

$$Y_a = Y_m \left[1 - K_y \left(1 - \frac{ET_a}{ET_c}\right)\right]$$

5.2 Inputs

- K_y **Values:** Stored in `residue_utils.py` (e.g., Cotton $K_y = 0.85$, Maize $K_y = 1.25$).
- ET **Ratio:** Currently supplied manually via the API parameter 'et_ratio'.

5.3 Status

This module is currently disabled by default (‘et_based=false‘) pending integration with real-time satellite evapotranspiration data

6 Case Study Validation: Yavatmal Survey No. 35

To validate the system, we ran the logic against a known plot in Khairgaon, Yavatmal.

6.1 Parameters

- **Survey No:** 35
- **Rainfall Anomaly:** +32.82% (No water stress, $ET_a/ET_c \approx 1.0$).

6.2 Results

The system selected the appropriate methods(similar to table 2) and yielded results consistent with manual scientific calculations.

Crop	**Source Selected**	**Method**	**Yield** (Y)	**Residue** (R)
Soybean	CCE (Farmers Avg)	Method A	1.04 t/ha	**2.317 t/ha**
Cotton	CCE (Lint Benchmark)	Method B	1.14 t/ha	**1.390 t/ha**
Gram	District APY	Method A	0.80 t/ha	**0.968 t/ha**
Sugarcane	District Productivity	Method C	75.0 t/ha	**21.75 t/ha**

Table 3: Results showing alignment between API outputs and manual models.

A Usage Manual

A.1 Product Overview

The **TerraStack Bio-Residue Estimation API** is a commercial-grade intelligence tool designed for biofuel companies, biomass aggregators, and sustainability consultants. It replaces manual field surveys with instant, scientifically validated residue availability data.

Key Value Propositions:

- **Precision:** Replaces regional guesses with plot-level calculations derived from satellite imagery.
- **Speed:** Reduces procurement planning time by up to 80% by enabling remote site assessment.
- **Compliance:** Uses IPCC 2019 standards accepted by major carbon registries.

A.2 Key Features

A.2.1 1. Point-Based Estimation

Function: Provide a latitude/longitude coordinate and a search radius (e.g., 5km), or a closed polygon to receive a comprehensive biomass audit.

A.2.2 2. Smart Yield Resolution

Function: The system automatically selects the most accurate data source available without user intervention.

Priority	Data Source	Accuracy Level
1st	Crop Cutting Experiments (CCE)	**High** (Field Measured)
2nd	District APY Statistics	**Medium** (Govt. Records)
3rd	State/National Averages	**Baseline** (Fallback)

Table 4: Automatic Data Selection Hierarchy

Transparency: Every API response includes a `yield_source_log`, explicitly stating the origin of the data (e.g., *"Source: CCE, Yavatmal District"*), ensuring auditability.

A.2.3 3. Visual Analytics

Function: The API generates professional-grade visualization assets for immediate inclusion in reports.

- **Residue Bar Chart:** Breakdown of biomass tonnage by crop type.
- **Monthly Harvest Timeline:** Visualizes exactly when feedstock will be available (e.g., "Peak availability: November-December").

A.3 Quick Start Guide

A.3.1 Step 1: Authentication

All requests require an API key passed in the header:

```
X-API-Key: your_assigned_key_here
```

A.3.2 Step 2: Basic API Call

Endpoint: `GET /api/v1/crop_yield/estimate`

```
GET /estimate?lat=20.165&lon=78.158&radius=1000&season=kharif
```

A.3.3 Step 3: Interpreting the Response

The JSON response contains three key sections:

1. **Summary:** Total area (ha) and total residue tonnage (t).
2. **Crop Breakdown:** Tonnage specific to each crop (e.g., *"Cotton: 351 tonnes"*).
3. **Visualization URLs:** Links to download the generated charts.

A.4 Integration Example (Python)

```python
import requests

def get_biomass_data(lat, lon):
    url = "https://api.terrastack.in/api/v1/crop_yield/estimate"
    headers = {"X-API-Key": "YOUR_KEY"}
    params = {
        "lat": lat,
        "lon": lon,
        "radius": 2000,
        "season": "kharif"
    }

    response = requests.get(url, headers=headers, params=params)
    return response.json()

# Example: Get data for a site in Yavatmal
data = get_biomass_data(20.165, 78.158)
print(f"Total Residue Available: {data['summary']['total_residue_tonnes']}
    tonnes")
```

Listing 1: Sample Python Client

A.5 FAQ

Q: How accurate are the estimates?
When CCE data is available (in supported districts like Yavatmal), accuracy is within ±15% of field measurements. In fallback regions utilizing APY data, accuracy is generally within ±25%.
Q: Can I get historical data?
Yes. The `year` parameter supports data retrieval for 2023/2024/2025, allowing for historical trend analysis.
Q: Is this data suitable for carbon auditing?
Yes. The system uses the **IPCC 2019 Refinement** methodology, which is the standard accepted by major carbon registries (Verra, Gold Standard) for calculating agricultural biomass.

B Appendix B: Sample Outputs & Data Logs

This section demonstrates the actual visual and data assets generated by the API for the Yavatmal case study (Survey No. 35).

B.1 B.1 Visual Analytics Dashboard

The `residue_grapher.py` module automatically renders high-resolution PNGs.

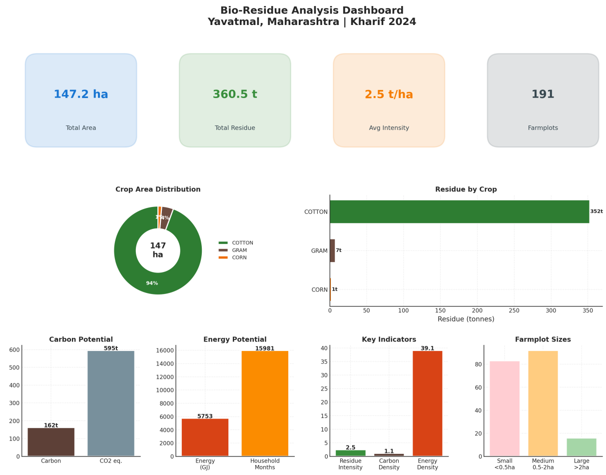

Figure 1: **Executive Summary Dashboard:** A high-level view combining total area, tonnage, and key efficiency metrics into a single view.

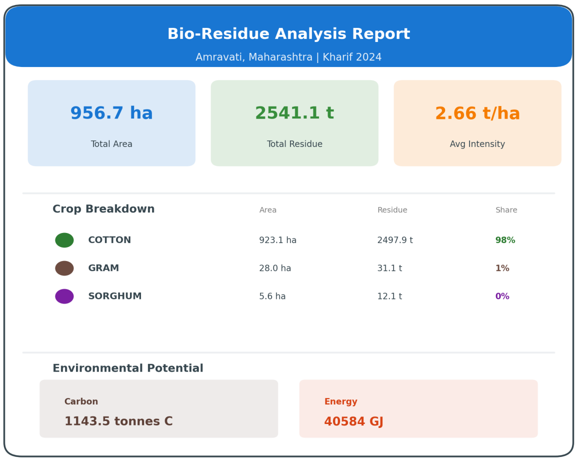

Figure 2: **Statistics Card:** Key Performance Indicators (KPIs) including total carbon potential.

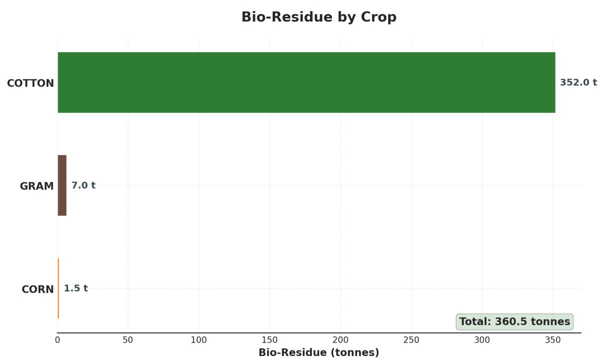

Figure 3: **Residue Volume by Crop:** Highlights Cotton as the primary feedstock source (352 tonnes) vs. Gram and Corn.

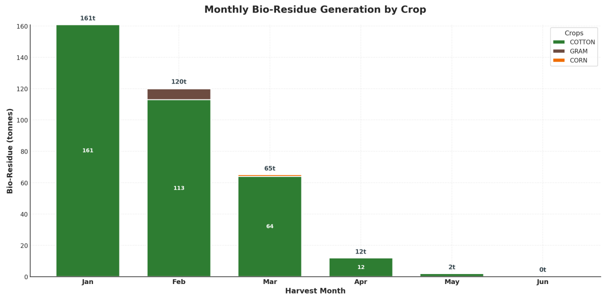

Figure 4: **Seasonal Availability:** Critical for logistics planning, showing peak residue generation in January (161t) and February (120t).

B.2 B.2 Granular Farmplot Logs (JSON)

The API provides deep transparency by returning data for every single identified plot. Below is a snippet from the `farmplots.json` output, showing pixel-level crop classification and yield source tracking.


```
{
  "count": 154,
  "farmplots": [
    {
      "farmplot_id": "7JHQ2G25+G438",
      "crops": {
        "COTTON": {
          "normalized_name": "cotton",
          "area_ha": 0.093,
          "yield_kg_ha": 448.3,
          "yield_source": "cce",  // Source: Crop Cutting Experiment
          "yield_method": "cce_spatial_query",
          "residue_per_ha_tonnes": 2.038,
          "residue_tonnes": 0.19,
          "harvest_month": 1,
          "confidence": 0.8758  // High confidence detection
        }
      },
      "total_residue_tonnes": 0.19
    },
    {
      "farmplot_id": "7JGQXGX5+JJVH",
      "crops": {
        "COTTON": {
          "normalized_name": "cotton",
          "area_ha": 0.908,
          "yield_kg_ha": 448.3,
          "yield_source": "cce",
          "residue_tonnes": 1.851,
          "harvest_month": 1,
          "confidence": 0.799
        }
      },
      "total_residue_tonnes": 1.851
    },
    {
      "farmplot_id": "7JHQ2G34+M3M6",
      "crops": {
        "COTTON": {
          "normalized_name": "cotton",
          "area_ha": 2.023,
          "yield_kg_ha": 448.3,
          "residue_tonnes": 4.122, // High volume plot
          "harvest_month": 2,
          "confidence": 0.7226
        }
      },
      "total_residue_tonnes": 4.122
    }
  ]
}
```


Listing 2: Sample Farmplot Log showing CCE-derived yield and confidence scores

Interpretation of Logs:

- **farmplot_id:** The unique code for the specific plot.
- **yield_source: "cce"**: Confirms that true data was used for this calculation.
- **confidence:** The satellite model's certainty (0.0 - 1.0) that the crop is correctly identified.
- **harvest_month:** Integer (1=Jan, 2=Feb) indicating when the residue will be physically available.

References

[1] IPCC (2019). *2019 Refinement to the 2006 IPCC Guidelines for National Greenhouse Gas Inventories*. Volume 4: Agriculture, Forestry and Other Land Use, Chapter 5: Cropland.

[2] Doorenbos, J. & Kassam, A.H. (1979). *Yield Response to Water*. FAO Irrigation and Drainage Paper No. 33.

[3] TerraStack Engineering Team (2026). *Bio-Residue Estimation System - Technical Documentation v1.2*.

[4] TerraStack Estimation Framework (2025). *Methodology for Bioresidue Estimation in Agricultural Plots: Case Study Yavatmal*.