IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 12, December 2025

DOI: 10.17148/IJARCCE.2025.1412130

IOT-Based Crop Recommendation System With Intrusion Detection

Tejas H R[1], Mr Yadhukrishna M R[2], Rakshitha K[3], Tejashwini N[4], Usha N[5]

Department of Information Science and Engineering, The Oxford College of Engineering,

Affiliated to Visvesvaraya Technological University, Belagavi, Karnataka, India[1-5]

Abstract: Agriculture remains the major occupation and source of livelihood for a large section of the population worldwide, especially in developing countries like India. However the sector faces challenges related to weather uncertainty, resource limitation, and lack of scientific decision-making tools. Most farmers usually decide which crop to grow based on guesswork or general government advice, but this information does not always match the exact conditions of their farmland, such as soil moisture, rainfall, and temperature, which can lead to low crop yield and poor income. To overcome these problems, this study presents a simple and low-cost IoT-based crop recommendation system that uses an ESP8266 microcontroller along with sensors like DHT11 for temperature and humidity, a soil moisture sensor, and a rainfall sensor to collect real-time data from the field and suggest suitable crops. In addition, a PIR sensor is used to detect the movement of wild animals or unauthorized people, helping to protect the crops and improve overall farm safety. Collected data processing is achieved by using a rule- based algorithm, extendable to machine learning models for recommending suitable crops and triggering alerts via a Flask- based web interface. This paper covers the system architecture, hardware-software integration, and experimental validation of the introduced prototype, discussing its potential to enhance agricultural productivity and sustainability.

Keywords: Internet of Things (IoT), Precision Agriculture, ESP8266, Crop Recommendation, Intrusion Detection, Smart Farming, Flask Framework.

I. INTRODUCTION

Agriculture is not just a job, it is a way of living for millions of people and it supports the whole world's economy. In countries like India, people's lives often depend on the farming seasons and harvest time. But farming is not easy nowadays. Farmers have to face many problems like unpredictable weather, climate changes, and lack of enough water and other resources, which makes their work very difficult and risky.

For a long time, farmers have been deciding what to grow and when to water their crops by using their own experience or advice passed down in their families, along with general tips given by the government. This old method is helpful, but it is not enough anymore because the climate keeps changing. Advice given for a whole area cannot match the exact condition of one small piece of land. Because of this, farmers may choose the wrong crops, use too much or too little water, and finally lose money.

This project starts with a very straightforward question: **How can we give nature a voice?** By integrating the Internet of Things into the field, it will translate environmental signals such as soil moisture and humidity into actionable insights. We propose a Smart Crop Recommendation System that not only monitors the field but also guides the farmer on what to grow. It also ensures security, using sensors to detect intruders and protect crops from animals and unauthorized entry

II. PROBLEM STATEMENT AND OBJECTIVE

Even with advances in agricultural technology, most farmers still work without real-time data. They are in a "blind spot," lacking information about their soil and immediate surroundings.

The Guesswork Game: Without exact information, farmers usually pick crops out of habit instead of choosing what suits the soil and climate, which results in poor yield.

The Resource Trap: Fear of drought makes farmers use too much water, which wastes water and spoils the soil.

The Security Gap: Even if a farmer grows a healthy crop, it can be destroyed by animals or stolen, and most smart systems do not focus on this safety problem.

Our Objectives: Our goal is to move beyond simple monitoring and create a system that acts as a decision- support partner for the farmer. Specifically, we aim to:

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⵋ Peer-reviewed & Refereed journal ⵋ Vol. 14, Issue 12, December 2025

DOI: 10.17148/IJARCCE.2025.1412130

- **Build a localized sensing node** using the affordable ESP8266 microcontroller to capture the unique "fingerprint" of the field's climate (temperature, humidity, rain, and soil moisture).
- **Translate raw data into decisions** by developing a backend engine that processes these signals and recommends the specific crop best suited for those exact conditions.
- **Secure the harvest** by integrating a Passive Infrared (PIR) sensor to detect and alert farmers to unwanted human or animal presence.
- **Democratize technology** by ensuring the entire system remains low-cost, easy to deploy, and accessible via a simple

III. LITERATURE REVIEW

A. Summary of Existing Work The agricultural sector has seen a paradigm shift with the introduction of "Smart Farming," where researchers have actively sought to replace intuition with data. The evolution of this field can be categorized into three distinct waves of innovation:

1) Sensor-Based Monitoring: Early research focused on proving that low-cost electronics could survive in the field. For instance, **Patil and Kale (2019)** developed a system using simple soil pH and moisture sensors . Their work was pivotal in demonstrating that microcontrollers could manage basic agricultural data, although their system was restricted to soil parameters and lacked environmental context like rainfall.

2) IoT and Cloud Visualization: As internet connectivity improved, the focus shifted to remote monitoring. **Singh et al. (2020)** proposed a framework utilizing DHT11 and soil sensors connected to an IoT cloud platform . This allowed farmers to visualize their field status remotely. However, while effective for observation, the system lacked a "brain"—it provided data but did not offer intelligent, actionable recommendations.

3) Machine Learning (ML) Predictions: Recognizing the need for better decision- making, researchers like Sharma et al. (2021) applied algorithms such as Random Forest and Decision Trees to agricultural datasets. These models achieved high accuracy in predicting suitable crops. Besides that, Gupta et al. (2021) checked the security aspect in farming by using PIR sensors to detect wildlife intrusion and hence created a critical need in crop protection.

B. Limitations of Current Systems Despite these advancements, a review of current technologies reveals several "blind spots" that prevent widespread adoption by ordinary farmers:

1) **Fragmented Sensing (The "Tunnel Vision" Problem):** Most existing solutions operate in silos. They either monitor the soil or the weather, but rarely both. For example, systems might measure soil pH but ignore rainfall patterns, leading to recommendations that are scientifically accurate but practically risky .

2) **The "Offline" Disconnect:** While Machine Learning models are powerful, they often rely on static, historical datasets rather than live field conditions . A model trained on last year's weather cannot help a farmer navigate a sudden drought today. This lack of real- time adaptability limits their practical effectiveness .

3) **Infrastructure Dependency:** Many IoT solutions are designed for areas with robust infrastructure, requiring continuous high- speed internet to function . In rural regions with spotty connectivity, these cloud- dependent systems often fail .

4) The Security Blind Spot: Perhaps the most overlooked limitation is physical security. While farmers invest heavily in growing crops, very few recommendation systems account for the risk of crop destruction by stray animals or intruders .

C. Identified Research Gap Synthesizing the limitations above, this project identifies and addresses three critical gaps in the existing body of knowledge:

1) The Multi-Parameter Integration Gap: There is a scarcity of systems that integrate a holistic set of variables - rainfall, humidity, temperature, and soil moisture into a single recommendation engine .Current systems lack this comprehensive environmental view

2) The Security-Plus-Management Gap: While pest detection and crop recommendation exist as separate fields of study, they are rarely combined. There is a clear need for a unified system that acts as both an agronomist (advising on crops) and a watchman (protecting the field) .

3) The Usability and Accessibility Gap: Many academic prototypes are complex and expensive, staying within the lab rather than reaching the field . There is a significant need for a farmer-centric solution that is affordable, easy to interpret, and capable of operating with minimal digital literacy.

IV. METHODOLOGY

A. System Design Overview

The system architecture is structured into four distinct functional layers:

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁂ Peer-reviewed & Refereed journal ⁂ Vol. 14, Issue 12, December 2025
DOI: 10.17148/IJARCCE.2025.1412130

1) Sensing Layer (The Senses): This layer consists of the physical sensors deployed in the field. Their job is to continuously "feel" the environment, capturing analog and digital signals related to soil and weather conditions.

2) Processing Layer (The Brain): At the edge, the ESP8266 microcontroller acts as the immediate processing unit. It collects raw data, performs essential cleanup (like noise filtering), and prepares the data for transmission.

3) Network Layer (The Nerves): This layer handles the flow of information. Using Wi- Fi, it transports the processed data from the field to the central server, utilizing standard protocols like HTTP or MQTT to ensure reliable delivery.

4) Application Layer (The Interface): This is where the machine speaks to the human. A Python-based backend analyses the data to generate crop recommendations, which are then displayed on a user-friendly Flask dashboard accessible via mobile or web.

B. Hardware components

The hardware selection was driven by three criteria: affordability, availability, and energy efficiency.

1) ESP8266 Microcontroller: We chose the ESP8266 because it is cheap and has built-in Wi-Fi. It works as the main unit that collects data from all the sensors and sends it to a server or cloud for further use.

2) DHT11/DHT22 Sensor: This sensor checks the temperature and humidity of the air. These values help in understanding the weather condition and deciding which crops can grow better in that climate.

3) Soil Moisture Sensor: This sensor measures how much water is present in the soil. It helps the system understand whether the land is better for water-loving crops like rice or for dry crops like millet.

4) Rainfall Sensor: This sensor detects precipitation intensity. By combining rainfall data with soil moisture readings, the system can make smarter irrigation recommendations, preventing water wastage.

5) PIR (Passive Infrared) Sensor: Addressing the security gap, the PIR sensor detects motion from animals or unauthorized persons. It adds a layer of protection, alerting farmers to potential crop damage or theft

C. Software components

The software ecosystem is designed to be lightweight and accessible.

1) Arduino IDE: We used the Arduino software to write and upload the program into the ESP8266. Through this, we control how the sensors work, collect real-time data from the field, and manage the Wi-Fi connection so the information can be sent to the cloud or server properly.

2) Python Environment: The backend of the system is built using Python, with libraries like Pandas and NumPy to handle and organize data. This setup processes the information coming from the sensors and runs the algorithms to suggest the most suitable crops for the field.

3) Flask Framework: We used Flask to create a web dashboard for the farmer. This simple framework shows real-time sensor readings, crop recommendations, and security alerts on a clear and responsive web page, making it easy for the farmer to monitor the field and take action quickly.

D. Data flow

The journey of a data point through our system follows a linear, five-step path:

1) Data Acquisition: The sensors (DHT, Soil, Rain, PIR) continuously monitor the physical environment, generating analog or digital signals.

2) Preprocessing (Edge): The ESP8266 reads sensor data, converts it into easy-to- understand values like soil moisture percentage, and removes electrical noise for accurate readings.

3) Transmission: The ESP8266 sends the data over Wi-Fi to the Flask server. It sends everything in JSON format using an HTTP POST request. This lets the server get the information quickly and use it right away.

4) Analysis & Decision: The Python backend reads the data it receives and runs it through the crop recommendation system. It checks the field conditions and compares them with stored crop details. Based on this, it decides which crop fits best. For example, if the soil is very moist and the temperature is high, it might suggest growing rice.

5) Visualization & Action: The farmer’s dashboard shows the live sensor readings, the crop suggestions, and any quick alerts from the PIR sensor. Everything is updated in real time so the farmer can see what’s happening immediately.

V. SYSTEM ARCHITECTURE

We made the system modular, so updating one part won’t affect the others. You can think of it as four layers that work together smoothly.

1) The Physical Layer: This is the hardware on the ground—the ESP8266 microcontroller connected to the sensor array. It’s designed to be rugged and energy-efficient.

International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⋇ Peer-reviewed & Refereed journal ⋇ Vol. 14, Issue 12, December 2025
DOI: 10.17148/IJARCCE.2025.1412130

2) The Edge Computing Layer: The ESP8266 handles some processing on its own before sending anything to the cloud. It cleans the sensor data and removes errors, so only correct information is sent.
3) The Communication Layer: This layer works like a bridge. It uses Wi-Fi to move the data from the field to the server. It's kept light so it can work even with slow internet in rural areas.
4) The Application Layer: This is the part that the user sees and uses. It's made with Python and Flask. It runs the crop recommendation system, saves the data, and shows the dashboard on the farmer's phone or computer.

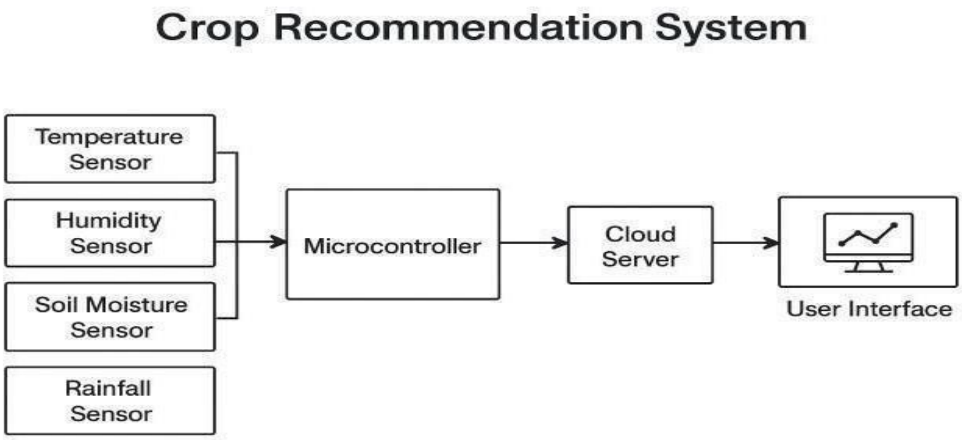

Fig. 1 System architecture

VI. CONCLUSION

This project started from the idea that farmers often have to take risks because they don't always know the conditions of their fields or the weather. We made this IoT-based crop recommendation system to show that technology can be simple and cheap but still useful. Even a small, low-cost system can give real-time data and advice, helping farmers make better choices and lower their risks.
We didn't just make graphs; we made a system that helps farmers decide what to plant. It uses a cheap ESP8266 and different sensors to collect real-time data and give useful advice to improve crops and manage the farm better. We also added a PIR sensor for security. During the day, the system helps with farming decisions, and at night, it alerts farmers about intruders or animals. Overall, it helps farmers save water, keep the soil healthy, and protect their fields. This shows that farming can be smarter when we use data to understand and respond to the land.

VII. FUTURE SCOPE

We see this prototype as just the beginning -a seed that has started to grow. There is still a lot of potential for it to develop into a complete smart farming system.

- **From Rules to Wisdom (Machine Learning):** Currently, our system uses strict rules to make decisions. The next step is to integrate advanced Machine Learning algorithms. This would allow the system to "learn" from previous harvests, becoming smarter and more accurate with every season.
- **Eyes in the Sky (Drone Integration):** While our ground sensors provide excellent micro-data, integrating aerial drones could provide a macro-view of crop health, spotting disease or irrigation issues that a single sensor point might miss.
- **Predictive Power (Weather APIs):** We plan to connect the system to external weather forecasting services. This would enable the system to warn farmers of incoming storms or droughts days in advance, rather than just reacting to current conditions.
- **Closing the Loop (Automated Irrigation):** Finally, the ultimate goal is automation. Future versions could directly control water pumps, automatically irrigating the field when the soil moisture drops below a specific threshold, ensuring not a single drop of water is wasted.

ACKNOWLEDGMENT

The authors thank the faculty and staff of the Department of Information Science and Engineering at The Oxford College of Engineering for their advice and assistance. The authors also acknowledge the open-source community for the IoT tools and frameworks used in this work and thank **Mr. Yadhukrishna M R** for his technical guidance and mentoring.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁘ Peer-reviewed & Refereed journal ⁘ Vol. 14, Issue 12, December 2025

DOI: 10.17148/IJARCCE.2025.1412130

REFERENCES

[1]. Z. Ali et al., "Sustainable crop recommendation using NPK sensors and humidity data", *Sustainability ,* vol. 12, no. 4, pp. 45–58, 2023.

[2]. M. Gupta et al., "Pest and wildlife detection in farms using PIR sensors and GSM alerts", *Journal of Field Robotics,* vol. 15, no. 2, pp. 112–125, 2021.

[3]. H. Kaur et al., "Hybrid IoT and machine learning recommendation system for adaptive crop suggestion", *Computers and Electronics in Agriculture,* vol. 198, pp. 107–119, 2022.

[4]. V. Kumar et al., "Precision farming with edge computing: Reducing internet dependency", *IEEE Access*, vol. 10, pp. 500–510, 2022.

[5]. S. S. Patil and S. A. Kale, "IoT crop recommendation using soil pH and moisture sensors", *International Journal of Electronics Engineering,* vol. 7, no. 1, pp. 22–29, 2019.

[6]. K. Roy et al., "Smart irrigation and crop suggestion system for water efficiency", *Advances in Intelligent Systems and Computing,* vol. 112, pp. 301–312, 2020.

[7]. A. Sharma et al., "Machine learning-based crop prediction using random forest and decision trees", *Journal of Agricultural Informatics,* vol. 12, no. 3, pp. 88–96, 2021.

[8]. R. Singh et al., "IoT cloud-based crop monitoring and real-time visualization", *Smart Farming Technologies*, vol. 5, no. 2, pp. 150–162, 2020.