Sustainable Precision Agriculture: An Autonomous Approach to Soil Sensing and Sapling Tree Managements

Tajwar Al Haque Robin (21101312), *Student, BRAC University,*
Khondaker Ashef Hossain (21101193), *Student, BRAC University,*
Fatin Ishtiaque (21101206), *Student, BRAC University,*
Alimun Al Mikdad (21301219), *Student Member, BRAC University,*
and Mr. Riad Ahemd, *Lecturer, BRAC University*

***Abstract*—Abstract — In modern urban forestry and precision agriculture, the successful adaptation of seedlings and saplings to evolving soil and microenvironmental conditions is crucial for their survival and growth. This research investigation introduces an autonomous agricultural robot that uses artificial intelligence (AI) and machine learning (ML) techniques to monitor soil quality and identify plant illnesses, with the goal of improving crop management and sustainable practices. The suggested system collects and processes real-time data on essential soil factors such as pH, conductivity, humidity, temperature, and nutrient levels. A normalization-based methodology is used to calculate a soil damage level (SDL), allowing for accurate assessment of soil health by comparing real-time data against predefined reference values for crops such as rose, guava, and tomato. The system's AI-based plant disease detection module, leveraging the YOLOv11 deep learning model, effectively classifies plant leaves into categories—healthy, scab, rust, and multiple disease— and generates corresponding health scores. The results of the study illustrate the system's capacity to identify soil deficiencies and diagnose plant diseases with high accuracy, allowing for early intervention and precision treatment techniques. The long-term objective is to optimize plant growth environments by addressing nutrient deficiencies and improving planting practices through the integration of AI-driven automation. This research contributes to the advancement of precision agriculture by providing a robust, autonomous system capable of delivering actionable insights for soil and crop management, ensuring enhanced productivity and sustainability . The ultimate goal is to optimize the growth environment by identifying and rectifying soil deficiencies and analyzing Soil Damage Level while generating a health score for the plants, ensuring the quality of saplings through precise planting methods.**

***Index Terms*— Precision Agriculture, Plant Disease Detection, Soil Quality Monitoring, Machine Learning, Artificial Intelligence, Autonomous Robot, YOLOv11.**

I. Introduction

BANGLADESH remains one of the most agriculture-intensive countries globally, characterized by extensive crop diversity and a large farming workforce. A cornerstone of modern agricultural management is soil health monitoring, which directly affects crop yield and resource utilization. Automated systems using multi-parameter soil sensors can measure pH, conductivity, humidity, temperature, and nutrient levels—factors integral to informed decision-making [1]. When integrated with AI-based models, such systems enable real-time alerts for anomalies in soil conditions, such as nutrient deficiencies or excessive temperature, thereby expediting countermeasures before significant crop damage occurs. Further, machine vision and deep learning methods enhance plant health diagnostics by detecting visual symptoms of disease on leaves, stems, and fruits. Early recognition of diseases—especially those manifesting through leaf discolorations or deformations—can lead to prompt treatments and significantly reduce yield losses (Khan et al., 2021; Sishodia et al., 2020) [2], [3]. Recent advancements in convolutional neural networks (CNNs), such as the YOLO (You Only Look Once) family of architectures, have made it feasible to identify plant diseases, pests, and nutrient deficiencies with high accuracy (Redmon Farhadi, 2018)[4]. By leveraging robust image processing modules, an agricultural robot can autonomously roam the field, capture images, and classify plant anomalies on-site. Additionally, the same platform can measure critical soil parameters—via sensors employing protocols like RS485—for immediate feedback on soil quality. The fusion of these hardware and software components closes the loop between problem detection and corrective action, an integral concept in precision agriculture (Liakos et al., 2018; Singh et al., 2020)[5], [6]. This research focuses on sapling tree management by creating an integrated framework that combines soil sensing, disease detection, and nutrient deficiency analysis to assist farmers and agronomists in making evidence-based decisions. Specifically, the system will monitor the soil conditions of newly planted saplings and detect potential foliar diseases at an early stage using image-based classification. By unifying these functionalities, farmers can react promptly to emerging threats such as suboptimal soil conditions or pathogen infections, ensuring a more robust and sustainable growth process. Ultimately, this holistic approach promises to improve yield, reduce resource wastage, and enhance environmental outcomes through judicious use of fertilizers, fertilizers, and water (Bhatnagar et al., 2022)[7]. This paper further aims to promote the development of specialized robotic solutions for Bangladeshi agriculture, thereby aligning with global sustainability agendas while strengthening local food security and economic resilience.

II. Related Works

The field of precision agriculture has seen extensive research and development in recent years, focusing primarily on automation and data-driven management techniques. Krishnaswamy and Bhat (2021)[8] explored the implementation of line-following robots in agricultural settings, highlighting their potential in reducing labor costs and increasing planting accuracy. Similarly, Kitić et al. (2022)[9] introduced 'AgRobot LALA', a real-time soil analysis robot that autonomously samples and analyzes soil nitrates, demonstrating the feasibility of integrating real-time data collection with agricultural operations. In the realm of soil sensing, Zhang, Liu, and Li (2019)[10] reviewed various robotic applications in precision agriculture, particularly emphasizing the role of multisensory systems in enhancing the accuracy of soil property assessments. Advances in machine learning, as discussed by Liakos et al. (2018)[5], have further augmented the capabilities of these systems, enabling more precise predictions and management decisions based on complex datasets. However, despite these advancements, the integration of such technologies at a larger scale, especially in resource-constrained settings, poses significant challenges. Cost and complexity issues are prevalent, as noted by Singh et al. (2020)[6], who call for more cost-effective and simpler solutions to make precision agriculture more accessible. Moreover, the work by Botta et al. (2022)[11] on robots, perception, and tasks in precision agriculture discusses the need for improved perception systems in agricultural robots to handle diverse environmental conditions effectively. This thesis aims to build upon these foundational studies by addressing some of these gaps, particularly focusing on cost-effectiveness and adaptability to diverse agricultural environments. The proposed system integrates autonomous navigation, real-time soil sensing, and data processing capabilities to offer a comprehensive solution for precision sapling management.

III. Methodologies

Precision agriculture leverages advanced technologies to enhance farming efficiency and sustainability. This research integrates YOLOv11-based plant disease detection with sensor-based soil health monitoring using an autonomous robotic platform equipped with a Pi camera and a 7-in-1 soil sensor. The system captures real-time plant and soil health data, processes it using deep learning and analytical algorithms, and provides actionable insights. It detects diseases (Healthy, Scab, Rust, Multiple Disease) and measures soil parameters (pH, conductivity, humidity, temperature, NPK levels). The robot autonomously navigates, collects, and stores data for further analysis, ensuring a holistic approach to precision farming. The integration of plant disease detection and soil monitoring is structured around a sequential and iterative workflow that ensures efficient data acquisition, processing, and decision-making. The following sections provide an in-depth discussion of the step-by-step process and how the methodologies interconnect.

System Initialization and Hardware Setup The system initialization process ensures seamless integration of the Plant Disease Detection System, Soil Monitoring System, and Robot Navigation System. The Raspberry Pi loads the trained YOLOv11 deep learning model for real-time plant disease detection, utilizing Python libraries such as Ultralytics, OpenCV, and NumPy for image processing. The Pi camera captures high-resolution images, ensuring accurate disease classification. The Soil Monitoring System establishes RS485 communication between the Arduino and the 7-in-1 soil sensor, enabling real-time collection of pH, conductivity, temperature, humidity, and NPK levels, with data stored in an Excel database for analysis. The Robot Navigation System initializes servo motors for precise soil sampling, while IR sensors facilitate autonomous path-following. Motor control logic is configured for smooth movement, allowing the robot to navigate agricultural fields, perform soil sampling, and collect critical data for precision farming.

Autonomous Navigation and Plant Disease Detection The robot autonomously navigates agricultural fields using IR sensors to detect and follow a designated path while adjusting its movement based on real-time sensor feedback. If an obstacle is detected, the robot temporarily halts, evaluates the situation, and resumes movement, ensuring uninterrupted operation. The Image Processing and Disease Detection system utilizes a Pi camera to capture live video frames, which are processed by the YOLOv11 deep learning model for real-time plant disease identification. Detected diseases are categorized, assigned a health score, and annotated with bounding boxes and confidence scores for visual representation. The Data Storage system logs disease detection results, including timestamps and location data, in an Excel database for further analysis. Additionally, automated alerts notify stakeholders if critical disease levels are detected, enabling timely intervention and effective crop health management.

Soil Sampling and Data Acquisition Upon reaching designated sampling locations, the robot autonomously halts and initiates the soil sampling process. A servo motor deploys the 7-in-1 soil sensor into the ground, while the Arduino retrieves pH, conductivity, humidity, temperature, and NPK levels. The collected data is processed and validated for accuracy, ensuring reliable soil health assessment. The Real-Time Analysis and Alerts system monitors soil parameters, triggering alerts if critical deviations occur, such as excessive temperature or low nitrogen levels. These alerts provide immediate feedback, enabling timely corrective actions. The Data Logging system continuously records sensor readings in an Excel database, while a Soil Damage Score is computed using normalization techniques, comparing real-time values against reference thresholds. This allows for trend analysis, soil health optimization, and data-driven agricultural decision-making.

Calculation of Soil Damage Level The Soil Damage Level (SDL) is quantified by comparing real-time sensor data with standard reference values. The process begins with deviation calculation, determining differences between measured and ideal soil parameters (pH, conductivity, humidity, etc.). These deviations are then normalized to ensure consistency across different units. A cumulative damage score is computed by aggregating normalized deviations, categorizing soil health into healthy, moderate damage, or severe damage. The SDL score is stored in an Excel database for trend analysis and decision-making. If the SDL exceeds critical thresholds, automated alerts are generated, enabling timely corrective actions to maintain soil health and prevent degradation.

$$S_{\text{damage}} = \sum_{i=1}^{n} \left| \frac{V_{\text{ref}}^{(i)} - V_{\text{measured}}^{(i)}}{V_{\text{ref}}^{(i)}} \right|$$

Data Visualization and Reporting The system provides real-time visual feedback and historical data insights to enhance agricultural decision-making. Live video feeds are annotated with plant disease detections, enabling continuous monitoring of crop health. Periodic reports are generated from the Excel database, summarizing soil health trends and highlighting changes over time. Additionally, graphical visualizations using Matplotlib present intuitive charts for long-term field analysis, allowing users to identify patterns, assess risks, and implement targeted interventions for improved crop and soil management. **Deployment and Future Improvements** The system is deployed in real agricultural environments to enable continuous monitoring of plant health and soil conditions. Future improvements include cloud-based integration for remote access to real-time data, enhancing decision-making efficiency. Model accuracy will be refined through expanded datasets and improved deep learning algorithms for better disease detection and soil analysis. Additionally, predictive analytics will be developed to forecast potential soil and plant health issues, allowing proactive interventions. These advancements will further enhance system capabilities, optimize agricultural productivity, and promote sustainable farming practices.

A. *Algorithm of the Entire System*

The system's functionality is driven by three core algorithms ensuring precision, efficiency, and reliability in soil monitoring and plant health assessment.

Soil Data Acquisition and Storage This algorithm retrieves real-time sensor data from an Arduino and processes key parameters such as pH, conductivity, humidity, temperature, and NPK levels. The data is stored in an Excel file, ensuring structured record-keeping for trend analysis and decision-making.

Reading Sensor Data and Saving to Excel – The system establishes serial communication with the Arduino, continuously parsing sensor readings and updating an Excel database for real-time tracking. Critical thresholds trigger warnings (e.g., high temperature or low nitrogen levels).

Calculation of Soil Damage Level (SDL) – Normalization techniques compare sensor readings with reference values to compute a damage score, categorizing soil health into healthy, moderate damage, or severe damage.

Warning Generation for Critical Parameters – Alerts are generated when soil parameters exceed predefined thresholds, enabling immediate corrective action to maintain optimal soil health.

Rose	Parameter	Ideal Amount	Warning Call	Message
1	Soil pH	6.5	<5	pH Low
2	Soil Conductivity	1000	<400	Conductivity Low
3	Soil Humidity	60%	<40%	Humidity Low
4	Soil Temperature	18°C	>40°C	Temperature High
5	Nitrogen	100	<40	Nitrogen Low
6	Phosphorus	35	<15	Phosphorus Low
7	Potassium	180	<100	Potassium Low

TABLE I: Soil Parameter Thresholds and Warning Levels for Rose Sapling

Guava	Parameter	Ideal Amount	Warning Call	Message
1	Soil pH	6.25	<5	pH Low
2	Soil Conductivity	1500	<800	Conductivity Low
3	Soil Humidity	70%	<40%	Humidity Low
4	Soil Temperature	25.5°C	>40°C	Temperature High
5	Nitrogen	15	<7	Nitrogen Low
6	Phosphorus	15	<5	Phosphorus Low
7	Potassium	200	<150	Potassium Low

TABLE II: Soil Parameter Thresholds and Warning Levels for Guava Sapling

Tomato	Parameter	Ideal Amount	Warning Call	Message
1	Soil pH	6.4	<5	pH Low
2	Soil Conductivity	2500	<900	Conductivity Low
3	Soil Humidity	70%	<50%	Humidity Low
4	Soil Temperature	21°C	>40°C	Temperature High
5	Nitrogen	200	<100	Nitrogen Low
6	Phosphorus	40	<30	Phosphorus Low
7	Potassium	250	<100	Potassium Low

TABLE III: Soil Parameter Thresholds and Warning Levels for Tomato Sapling

Excel File Initialization – The system ensures structured data storage by initializing an Excel workbook with predefined column headers, allowing seamless data logging over time.

Integration of Algorithmic Steps The system workflow begins with Excel file setup (Step 4), followed by real-time data acquisition (Step 1). Soil Damage Level calculations (Step 2) provide a holistic soil health assessment, while automated alerts (Step 3) enable proactive interventions. Together, these steps form a scalable, automated soil monitoring framework, enhancing precision agriculture and sustainable farming practices.

Step 5: Continuous Monitoring and Parsing of Sensor Data Efficient monitoring of soil health parameters is critical for sustainable agricultural practices and precision farming. This system utilizes the RS485 communication protocol to interface with soil sensors, acquiring essential parameters such as pH, electrical conductivity, humidity, temperature, and nutrient levels (NPK). The methodology outlined in this section integrates multiple subsystems to ensure reliable communication, precise data acquisition, and systematic processing, while addressing challenges such as data integrity, communication errors, and system scalability.

B. *Algorithm: Plant Disease Detection System*

This system is built to revolutionize precision agriculture by utilizing deep learning to enable automated and accurate plant disease detection. Leveraging the YOLOv11 (You Only Look Once) model, it identifies plant diseases with high precision, ensuring versatility across various agricultural settings. A key feature of the system is its real-time analysis capability, achieved through seamless video stream processing and batch image analysis, allowing immediate feedback on plant health. This empowers farmers and agricultural professionals to take prompt actions to mitigate the spread of diseases and reduce crop loss. The system includes a decision-support framework, which provides actionable insights through health scoring mechanisms. These scores quantify the severity of detected diseases, supporting informed decision-making for targeted interventions. Designed to be scalable, the system can adapt to various agricultural environments and applications. It incorporates efficient processing and detailed logging mechanisms, ensuring transparency and traceability of results.

1) Step 1: Training the Model: Data Preparation: To train the model, a custom dataset is curated by collecting a large set of plant images and annotating them with relevant disease labels. This dataset is then split into three subsets to ensure a balanced approach to training and evaluation: 80% of the images are used for training, 10% for testing, and the remaining 10% for validation. To further enhance the dataset's diversity and improve the model's ability to generalize across different conditions, data augmentation techniques are applied. These include transformations such as scaling, rotation, and brightness adjustments, which simulate real-world variations and make the model more robust to environmental factors. Model Setup: Ultralytics is used as the package manager for easy installation and configuration of the YOLOv11 model. After initialization, training parameters, such as batch size, learning rate, and the number of epochs, are carefully chosen to optimize the model's performance. Training Process: Training involves two sessions to optimize the model's performance. The first session is conducted for 50 epochs with a batch size of 32, while the second session extends to 100 epochs with a batch size of 16. This dual-session approach allows for a comprehensive assessment of the model's performance, evaluating accuracy rates, loss values, and mean average precision (mAP). The model is specifically trained to classify four plant disease categories: Healthy, Multiple Disease, Rust, and Scab. Upon completion of the training, the model generates a weight file along with confusion matrices, which are used for further evaluation of its accuracy and performance.

2) Step 2: Model Initialization and Loading: YOLOv11 operates by dividing an image into regions, predicting bounding boxes, and computing probabilities for each region. The model consists of three components: head, neck, and body. The head predicts multiple boxes per grid cell, and non-max suppression selects the best predictions. The neck improves detection accuracy using PANet, and the body utilizes a modified Cross Stage Partial (CSP) network for feature extraction. The pre-trained YOLOv11 model is initialized using the weight file (best.pt) and relevant dependencies, such as ultralytics.YOLO, are imported for smooth model setup.

3) Step 3: Batched Image Inference: This process involves high-throughput processing of plant images in batches. The trained YOLOv11 model analyzes each image to detect and classify plant diseases. Detection results for each image are stored for further processing, enabling efficient disease identification across large-scale agricultural environments.

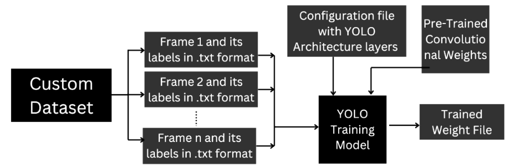

Fig. 1: YOLOv11 Model Working

4) Step 4: Detection Results Processing: Detection results are processed by extracting essential information like: Disease class (e.g., Rust, Scab), confidence scores, bounding box coordinates. For each detection, the class ID is mapped to its corresponding disease label. Health scores are calculated based on the disease's severity, providing valuable insights for agricultural professionals.

5) Step 5: Confidence Thresholding: This step filters out detections with confidence scores lower than a predefined threshold (e.g., 0.3). Detections above the threshold are retained, ensuring that only reliable results are considered for further processing. False positives are discarded and logged for future analysis, improving the overall accuracy and reliability of the system.

6) Step 6: Calculate Health Score: A numerical health score is calculated for each plant based on the severity of the detected disease. For example, mild diseases may receive a higher health score, severe diseases are assigned lower scores. This health scoring system provides actionable data for prioritizing interventions and monitoring the overall crop health.

$$\text{Health Score} = \begin{cases} 4, & \text{Healthy} \\ 3, & \text{Scab} \\ 2, & \text{Rust} \\ 1, & \text{Multiple Disease} \\ 0, & \text{Unknown} \end{cases}$$

7) Step 7: Logging Detected Conditions: All detection results, including disease labels, confidence scores, and health scores, are logged. This data is crucial for tracking disease patterns, making informed decisions, and ensuring transparency. If no disease is detected, an appropriate message is displayed.

8) Summary of Integrated Algorithms:: The plant disease detection system uses a combination of algorithms and workflows to ensure high accuracy and efficiency. Starting with the initialization of the YOLOv11 model, followed by image inference, detection processing, confidence thresholding, and health scoring, the system ultimately logs and displays results for further action. These integrated processes ensure that the system is well-suited for the demands of precision agriculture, providing a valuable tool for early disease detection and intervention.

Step	Algorithm Name	Purpose
1	Training the Model	Develop an accurate plant disease detection system.
2	Model Initialization and Loading	Load the YOLO model for inference.
3	Batched Image Inference	Process multiple images for detection.
4	Detection Results Processing	Generate insights from detection results.
5	Confidence Thresholding	Filter out detections with low confidence.
6	Calculate Health Score	Assign health scores based on detection results.
7	Logging Detected Condition	Save or display detected diseases.

TABLE IV: Summary of Integrated Algorithms

C. Algorithm: Robot Navigation for Autonomous Soil Sampling and Data Collection

This module aims to enable the robot to navigate autonomously using infrared (IR) sensors, ensuring it can detect and follow a predefined path while making real-time decisions to stay on track. The robot utilizes a set of three IR sensors (left, right, and middle) to detect the line on the ground and adjust its movements accordingly. By interpreting signals from these sensors, the robot can move forward, turn, or stop as needed. This capability allows the robot to autonomously navigate toward specific locations, such as soil sampling points, where it can pause, collect soil data, and continue its journey. This autonomous navigation capability is fundamental to enabling the robot to perform its tasks in dynamic environments, ensuring accurate and efficient soil sampling. The algorithm consists of several distinct steps designed to guide the robot's movements and actions. The process begins by reading IR sensor values to detect the line on the ground. Depending on the readings from the left, middle, and right sensors, the robot adjusts its movement. If both the left and right sensors detect the line while the middle sensor does not, the robot stops to deploy the soil collection mechanism, gathers data, and then resumes movement. If none of the sensors detect the line, the robot continues to move forward. If only one of the left or right sensors detects the line, the robot will turn accordingly to stay on track. The dynamic adjustment of turning speed based on sensor feedback ensures smooth and efficient navigation. In addition to navigation, the robot incorporates a soil sampling and data acquisition module. When the robot reaches a predefined soil sampling location, it stops, deploys the servo mechanism to position the 7-in-1 soil sensor, and collects critical soil data such as pH, conductivity, humidity, temperature, and NPK values. Once the data is collected, the servo mechanism is retracted, and the robot resumes its movement. This process ensures the robot effectively collects soil data while minimizing disruption to its path and tasks. The system also includes a movement control module that manages the robot's movement via the control of DC motors. The robot can move forward, turn left, turn right, or stop, based on input from the IR sensors. The movement is controlled by activating specific motor pins for forward and backward motion, while the speed of movement is adjusted using an analog Write function. This allows for precise control of the robot's movement and the ability to make accurate turns and stops as needed. The integration of the navigation and soil sampling modules allows the robot to operate autonomously in real-world environments. The robot follows the predefined path using its IR sensors and stops at sampling points to collect soil data. Upon completing the data collection, the robot resumes its movement and continues the process. This integration ensures that the robot can navigate autonomously and perform its task of soil sampling without human intervention, providing continuous data collection for real-time insights. The research integrates plant disease detection and soil monitoring through advanced deep learning and sensor-based systems. The robot's ability to autonomously navigate and collect soil data, combined with the plant health detection system, enables efficient and proactive decision-making in agriculture. This system represents a significant advancement in autonomous robotic systems for precision agriculture, facilitating sustainable and data-driven agricultural practices. This work demonstrates the potential for robotics in agricultural applications, particularly in improving efficiency, accuracy, and sustainability in soil analysis and plant health monitoring. By leveraging autonomous navigation, real-time sensor feedback, and deep learning algorithms, this system provides a comprehensive solution for modern agricultural challenges. This innovative approach can contribute to the development of smarter farming practices, ultimately enhancing productivity while minimizing environmental impact.

IV. Design and Development of the Autonomous Agricultural Robot

The development of an autonomous agricultural robot designed for precision agriculture involves intricate planning and integration of various technologies. This section delves into the system architecture, key components, and navigation techniques that collectively enable the robot to perform soil sensing and management tasks efficiently.

A. System Architecture

The autonomous agricultural robot's system architecture is designed to support robust operation in diverse farming environments. It consists of three primary layers: the sensing layer, control layer, and actuation layer. The sensing layer includes various sensors for soil analysis and environmental monitoring, such as pH, moisture, and NPK sensors, which collect real-time data critical for precise agriculture practices. The control layer features a microcontroller or embedded system that processes sensor data, handles navigation algorithms, and makes decisions based on predefined criteria. Lastly, the actuation layer comprises motors and mechanical structures that facilitate movement and operational tasks like sampling and data logging.

B. Hardware Components

Raspberry Pi 5 as the robot's central processing unit, managing complex computations and running the YOLOv11-based plant disease detection model.
Arduino Uno is the robot's primary microcontroller, handling sensor data collection and actuator control.

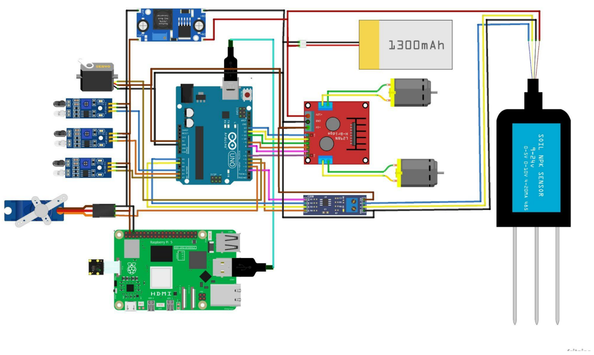

Fig. 2: Circuit Diagram of the Autonomous Agricultural Robot

DC-DC buck converter steps down the voltage from the primary battery to suitable levels for different components.
12V (3S) Lithium Polymer (LiPo) battery with a capacity of 2800mAh and a discharge rate of 30C to power the Robot.
Infrared (IR) sensors enable the robot to autonomously navigate by detecting field markers and avoiding obstacles.
12V DC motors with 1.5 Nm torque and 150 RPM speed, driven by the L298N motor driver to assist in forward/backward and turning left/right. Servo Actuator for deploying and retracting the soil sensor mechanism, ensuring accurate positioning for sampling.
RS485 module enables robust, long-distance data transmission between the Arduino and soil sensor.
7-in-1 Soil Sensor measures soil pH, moisture, NPK levels, temperature, and conductivity.
12 MP Pi Camera captures high-resolution images for real-time disease detection using the YOLOv11 model.
L298N motor driver controls the robot's two DC motors, providing bidirectional movement and speed regulation.

C. Robot Navigation

The navigation system of the robot is pivotal for its operation in agricultural fields. It utilizes a combination of line-following techniques and GPS for precise movement and positioning. The line-following system, primarily based on optical sensors, detects marked paths between crop rows, allowing the robot to move along these lines accurately. For areas lacking physical line guides, GPS coordinates are used to navigate across broader fields. The integration of ultrasonic sensors helps in obstacle detection, enabling the robot to avoid unexpected barriers like rocks, tools, or uneven terrain, thus ensuring continuous operation without manual intervention. The robot is navigated autonomously using a Line Following Algorithm (LFR) combined with infrared (IR) sensors. These sensors detect a predefined path on the ground, ensuring precise movement. The motor driver controls the wheels, adjusting direction based on sensor inputs. When the robot reaches a sampling point, it halts, deploys the servo mechanism, and collects soil data using RS485 communication with the 7-in-1 soil sensor. The gathered data is then transmitted to the Raspberry Pi for processing and storage. This ensures efficient navigation, accurate soil sampling, and real-time analysis for precision farming.

V. Result

Expected Results and Performance Metrics from Detection

The evaluation of the plant disease detection system is based on the performance metrics derived from a confusion matrix, which compares predicted outcomes with actual results. The confusion matrix categorizes plant leaves into five types: Healthy, Multiple Disease, Rust, Scab, and Background, offering a comprehensive overview of the model's classification accuracy. The model performed exceptionally well in detecting Rust and Scab diseases, with correct classifications of 62 Rust and 56 Scab samples. However, challenges arose with the "Multiple Disease" category, which proved difficult due to its similar patterns and limited representation in the testing dataset. The model demonstrated a strong ability to detect Healthy plants, correctly identifying 94 instances, but struggled with background separation, misidentifying healthy and infected leaves as part of the background. This indicates a need for further refinement in the model's ability to distinguish between background elements and actual plant diseases. The use of F1-score, precision, recall, and accuracy metrics highlighted the model's overall efficacy, with stronger results when detecting actual disease conditions compared to background contamination. The analysis also revealed areas for improvement, such as feature enhancement and improved data labeling methods, which are expected to increase the model's accuracy. Ultimately, the model shows promise for application in precision agriculture, especially for the automatic detection of Rust and Scab diseases, though improvements are needed to handle the complexities of multiple diseases and background noise. The performance of the model is illustrated in the confusion matrix (Figure 3).

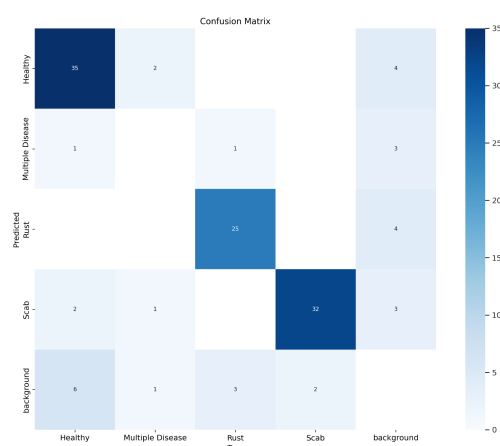

Fig. 3: Confusion Matrix

A. Normalized Confusion Matrix Analysis

The normalized confusion matrix provides insights into the detection rates for various classes by displaying detection accuracy across different categories rather than raw counts. The model showed impressive performance, correctly identifying 94% of Scab cases and 86% of Rust cases. In contrast, the ”Healthy” category had a lower recall rate of 80%. The model exhibited difficulty in accurately classifying multiple diseases, with only 0.03% positive findings for this class, often confusing them with Scab or Background. This analysis revealed that the model's struggles lie in distinguishing between multiple plant diseases and separating these diseases from background elements. While the model successfully identified Healthy, Rust, and Scab leaves with accuracies of 98%, 95%, and 95%, respectively, it faced challenges in correctly identifying the Multiple Disease and Background categories. These issues highlight the need for refined data augmentation and model tuning to address the difficulties in recognizing similar disease patterns and distinguishing between plant leaves and their backgrounds. To improve performance, future efforts should focus on enhancing medical data and feature design, along with fine-tuning the decision-making parameters of the model.

Predictions Confidence

Prediction confidence refers to the model's certainty in its predictions, with higher values signifying stronger confidence. The YOLOv11 model demonstrates a range of confidence levels, from 0.3 for lower confidence predictions to 0.9 for higher certainty. In cases of rust disease detection, the model showed lower confidence, with values around 0.21, whereas for Scab, the model demonstrated high confidence with values approaching 0.90. The model's confidence was also evaluated on various labeled data, with average confidence levels ranging from 0.5 to 0.9, which indicates reliability in its predictions. Although the model performed well for Scab and Healthy leaves, the detection of rust disease needs improvement due to lower confidence scores in that category.

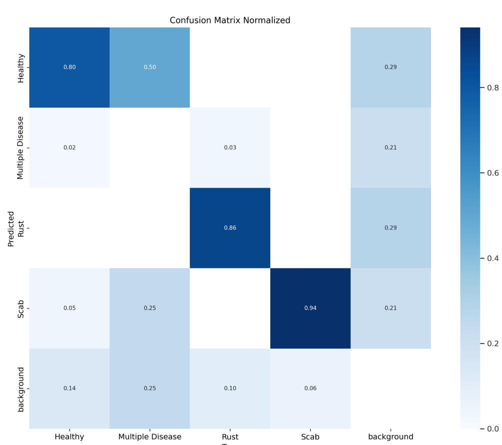

Fig. 4: Normalized Confusion Matrix

Training Loss Graphs

Training loss graphs illustrate the progress of the YOLOv11 model over training epochs, showing improvements in both


(a) Healthy (b) Rust (c) Scab

Fig. 5: Prediction Confidence of YOLOv11 for (a) Healthy, (b) Rust, and (c) Scab

object classification and bounding box regression. The training loss graphs demonstrate a steady decrease in the loss values, indicating that the model is improving its ability to learn from the training data. Key metrics, such as Train/dfl loss, show a downward trend, suggesting better perception in bounding box predictions. Additionally, validation losses, including val/cls and val/dfl loss, confirm that the model is generalizing well to unseen data, leading to more accurate predictions. The evaluation of mAP (mean average precision) metrics revealed consistent improvement during training, further confirming the model's progress in object detection and classification. These results demonstrate that the YOLOv11 model is becoming increasingly effective at identifying plant diseases and bounding box detections.

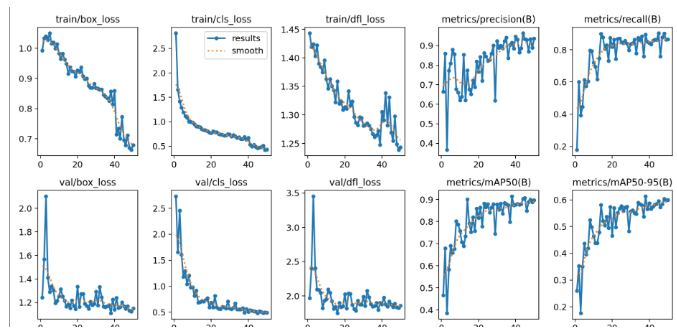

Fig. 6: Train Loss Graphs

YOLOv11 Detection in Agricultural Trials

The real-time detection system was tested under agricultural field conditions, showing promising results in detecting plant diseases with bounding boxes and confidence scores displayed in processed frames. The detection accuracy was found to be consistently high, with confidence levels typically above 0.5, indicating reliable disease classification. Healthy plants were detected with 74% confidence, and the detection system successfully avoided weak predictions below 0.3, ensuring reliable results. However, certain challenges were noted, including difficulties with lighting conditions and network speed instability, which led to occasional artifacts in real-time processing. The model struggled with distinguishing between diseases such as Rust and Scab due to their visual similarities, which is an area requiring future improvements in data processing and model differentiation capabilities.

Health Score Interpretation

A health scoring system was developed to assess plant conditions based on disease detection. Healthy plants received a score of 4, while plants with diseases like Rust, Multiple Disease, and Scab received lower scores. The system demonstrated high performance, processing video streams with minimal delay (90–100 ms per frame) and reliably identifying plant conditions in real time. The system's high reliability and performance were evident in its ability to process video frames rapidly, offering accurate results with confidence levels exceeding 0.7. However, network instability and poor lighting conditions led to occasional misclassifications and reduced detection accuracy. The future development plans include enhancing the system's robustness by addressing these issues, integrating diverse plant types, and implementing a web-based monitoring and control interface for broader agricultural applications.

These results and performance metrics underscore the promising capabilities of the YOLOv11-based plant disease detection system. The model shows significant potential for real-time agricultural applications, particularly for detecting diseases like Rust and Scab. However, further advancements in data preprocessing, feature design, and model tuning will be necessary to handle more complex disease categories and environmental factors. As future research continues, these improvements will enhance the system's accuracy and usability, making it a valuable tool for precision agriculture.

Soil Analysis Results

This study aimed to evaluate key soil parameters such as pH, conductivity, humidity, temperature, and macronutrient levels (Nitrogen, Phosphorus, Potassium) using an autonomous agricultural robot. The robot was deployed across various field locations, autonomously collecting data to calculate the Soil Damage Level (SDL) by comparing it to reference thresholds for rose, guava, and tomato crops. The analysis revealed soil variances, identifying areas with pH imbalances and nutrient deficiencies. For example, some fields had low Nitrogen levels, suggesting the need for fertilization, while others had excess Phosphorus, requiring adjustments to avoid toxicity. These findings provide actionable insights for crop-specific soil treatments, ensuring improved soil health and sustainable farming practices. The robot's real-time data collection enables precise, timely interventions to optimize soil fertility and crop yield. Soil Readings of a Sapling Rose, Guava, Tomato.

Intake	pH	Cond.	Hum.	Temp (°C)	N	P	K
1	7.00	17.7	10.0	23.0	12	17	35
1	6.58	41.3	11.9	23.0	29	41	82
2	3.00	55.9	15.5	23.0	39	55	111
2	3.00	56.0	15.6	23.0	40	56	112

TABLE V: Raw Data Collected from Rose Sapling Trees (Before Using Fertilizer)

Intake	pH	Cond.	Hum.	Temp (°C)	N	P	K
1	6.61	74.4	11.7	21.8	53	74	148
1	6.50	74.4	11.9	21.8	53	74	148
2	6.44	59.9	9.8	21.8	42	59	119
2	6.42	60.0	10.0	21.8	42	60	120

TABLE VI: Raw Data from Guava Sapling Trees Before Using Fertilizer

Intake	pH	Cond.	Hum.	Temp (°C)	N	P	K
1	8.57	32.3	11.0	22.6	23	32	64
1	6.80	36.6	12.8	22.6	26	36	73
2	6.64	37.3	12.7	22.6	26	37	74
2	6.54	38.0	12.7	22.6	27	38	76

TABLE VII: Raw Data from Sapling Trees Tomato Before Fertilizer Application

After Using Fertilizer:

Intake	pH	Cond.	Hum.	Temp (°C)	N	P	K
1	8.12	85.2	15.8	22.9	60	85	170
1	6.76	96.1	21.2	22.9	68	96	192
2	6.53	103.7	21.8	22.9	71	99	198
2	6.51	98.6	21.8	22.9	70	98	197

TABLE VIII: Raw Data Collected from Sapling Trees (After Using Fertilizer)

Intake	pH	Cond.	Hum.	Temp (°C)	N	P	K
1	6.62	127.6	20.8	21.8	91	127	255
1	6.52	131.3	21.2	21.8	93	131	262
2	6.75	127.6	19.1	21.8	91	127	255
2	6.79	124.1	19.0	21.8	88	124	248

TABLE IX: Raw Data Collected from Sapling Trees (After Using Fertilizer)

Intake	pH	Cond.	Hum.	Temp (°C)	N	P	K
1	7.76	135.3	21.0	22.5	96	135	270
1	6.67	151.4	23.4	22.5	108	151	302
2	6.45	153.0	23.5	22.5	109	153	306
2	6.38	158.0	23.5	22.5	112	158	316

TABLE X: Raw Data Collected from Sapling Trees (After Fertilizer Application)

Comparative Analysis of Fertilizer Application on Rose, Guava, and Tomato Sapling A comparative analysis of soil parameters for rose, guava, and tomato saplings before and after fertilizer application reveals significant improvements in nutrient levels, soil health, and plant growth. Following fertilizer application, all three plant types exhibited enhanced nutrient availability, improved soil structure, and better moisture retention. The data indicated an increase in key macronutrients, such as nitrogen, phosphorus, and potassium, alongside a more balanced pH level. For rose plants, the fertilizer application led to a marked reduction in soil damage levels, signifying better soil structure and nutrient distribution, which promoted healthier growth. Similar improvements were observed in guava and tomato saplings, with noticeable increases in nitrogen and phosphorus levels, contributing to improved plant development. The fertilizer's role in optimizing soil conditions helped create an environment conducive to stronger root systems and better overall plant health for all three species. These findings highlight the positive impact of fertilizer application on soil fertility and agricultural productivity, benefiting a wide range of crops. **Soil Damage Level Analysis**

Intakes	Guava (SDL)	Tomato (SDL)	Rose (SDL)
Intake 1	0.9165	1.3861	2.2000
Intake 1	0.8185	0.9443	2.0841
Intake 2	0.7933	0.9216	2.0810
Intake 2	0.7438	0.9912	1.7721

TABLE XI: Raw Data Collected Directly from Sapling Trees (Before Using the Fertilizer)

Prior to fertilization, SDL values ranged from 0.74 to 2.19, with higher values indicating significant deviations from optimal soil health. The high SDL values observed in certain locations corresponded to extreme pH levels and nutrient deficiencies, suggesting the need for corrective measures such as soil amendments and optimized fertilization strategies.

Intakes	Guava (SDL)	Tomato (SDL)	Rose (SDL)
Intake 1	0.7594	0.8871	1.8505
Intake 1	0.6876	0.6919	1.5099
Intake 2	0.7282	0.8729	1.4756
Intake 2	0.7432	0.8749	1.4580

TABLE XII: Raw Data Collected Directly from Sapling Trees (After Using the Fertilizer)

Comparative Analysis The comparative analysis of soil damage levels (SDL) before and after fertilizer application shows significant improvement in soil health. Post-fertilization, SDL values decreased across all sapling types, ranging from 0.65 to 1.85, indicating better soil conditions and more balanced nutrient levels. This suggests that fertilizer effectively mitigated soil damage, corrected nutrient imbalances, and stabilized soil health. Regular monitoring and strategic nutrient management are necessary to maintain and further improve soil quality.

VI. Conclusion

This study highlights the integration of artificial intelligence and robotics in agriculture, demonstrating that autonomous farming robots equipped with deep learning and advanced sensors improve plant disease detection and soil health monitoring. The robot uses a Raspberry Pi, Arduino, and a soil detection system to assess soil properties and measure soil damage levels, enabling early identification of issues. The system also detects plant health through YOLO deep learning, helping farmers take quick action. While the technology shows promise, future improvements include better handling of tough surfaces, enhanced data connectivity, and broader environmental data collection. Overall, this AI-driven approach can enhance crop yield and sustainability, offering significant potential for modern agriculture.

Acknowledgment

The authors would like to thank Mr. Riad Ahmed, Lecturer at BRAC University for the supervision of this entire thesis.

References

[1] A. Ahmed, "Architecture of densenet-121," *OpenGenus IQ: Learn Algorithms, DL, System Design*, 2021, available from https://iq.opengenus.org/densenet-121-architecture/.

[2] M. A. Khan, N. Hussain, and M. Farooq, "Plant disease detection: Recent trends and challenges," *Sensors*, vol. 21, p. 3295, 2021.

[3] R. P. Sishodia, R. L. Ray, and S. K. Singh, "Applications of remote sensing in precision agriculture: A review," *Remote Sensing*, vol. 12, no. 19, p. 3136, 2020.

[4] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," 2018, arXiv preprint arXiv:1804.02767. Available from https://arxiv.org/abs/1804.02767.

[5] K. G. Liakos, P. Busato, D. Moshou, S. Pearson, and D. Bochtis, "Machine learning in agriculture: A review," *Sensors*, vol. 18, p. 2674, 2018.

[6] R. Singh, M. Jain, and P. Sharma, "Precision agriculture: Adoption and impact on agricultural productivity," *Agricultural Research Journal*, vol. 57, no. 2, pp. 89–102, 2020.

[7] S. Bhatnagar, A. Kumar, and R. Singh, "Smart farming: A step towards sustainable agriculture using artificial intelligence and the internet of things (iot)," *Journal of Agricultural Sciences*, vol. 10, pp. 45–57, 2022, available from https://www.agrijournal.org/.

[8] S. Krishnaswamy and R. Bhat, "Implementation of line following for agricultural robots," *Journal of Agricultural Robotics*, vol. 33, pp. 44–52, 2021.

[9] G. Kiti'c, D. Krkljes, M. Pani'c, C. Petes, S. Birgermajer, and V. Crnojevi'c, "Agrobot lala—an autonomous robotic system for real-time, in-field soil sampling, and analysis of nitrates," *Sensors*, vol. 22, p. 4207, 2022.

[10] X. Zhang, G. Liu, and Y. Li, "Robotic applications in precision agriculture: A review," *Agricultural Engineering International: CIGR Journal*, vol. 21, no. 3, pp. 1–13, 2019.

[11] A. Botta, P. Cavallone, L. Baglieri, G. Colucci, L. Tagliavini, and G. Quaglia, "A review of robots, perception, and tasks in precision agriculture," *Applied Mechanics*, vol. 3, pp. 830–854, 2022.

AUTHOR BIOGRAPHIES

Mr. Riad Ahmed completed the B.Sc degree from the Department of Robotics and Mechatronics Engineering, University of Dhaka, Bangladesh. He has been working as a Lecturer in the Computer Science and Engineering department at BRAC University, Bangladesh. Before that he was working as a machine learning engineer at Headless Technologies Limited and a Computer Vision Engineer at Pranon Global Limited. On top of that, he also worked as a Research assistant at the University of Dhaka. He supervised this research project, providing guidance on the integration of technological solutions in agriculture and ensuring the scientific rigor of the study.

Tajwar Al Haque Robin (21101312) is an undergraduate student at BRAC University pursuing a degree in Computer Science and Engineering. His academic interests lie in data science, the Internet of Things (IoT), and their applications in real-world problem-solving. He is particularly passionate about leveraging IoT and data analytics to develop intelligent systems for automation and predictive analytics.

Khondaker Ashef Hossain (21101193) is currently studying Computer Science and Engineering at BRAC University. His areas of interest are robotics, automation, machine learning, data analysis, and the application of technology in practical practices.

Fatin Ishtiaque (21101206) is an undergraduate student in the Department of Computer Science and Engineering at BRAC University. He focuses on software development, data science, and the integration of technology with environmental monitoring systems.

Alimun Al Mikdad (21301219) is a student at BRAC University, majoring in Computer Science and Engineering. His academic interests lie in robotics, automation, and the development of innovative technologies for agricultural advancements.