## A REPORT ON

**IoT-Based Leaf Disease Detection Using ESP32-CAM and TCS3200 with Telegram Alerts**

***Submitted by,***

**MANU M -20221LCS0028**

**C VINITH -20221LCS0029**

**SHRIJO GORAL -20221LCS0030**

**LOHITH N -20221LCS0019**

### *Under the guidance of,*

**Dr Naveen N M**

***In partial fulfillment of the award of the degree of***

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**At**



**PRESIDENCY UNIVERSITY**

**BENGALURU**

**MAY 2025**

**PRESIDENCY UNIVERSITY**

**PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

**CERTIFICATE**

This is to certify that the Internship/Project report **“IoT-Based Leaf Disease Detection Using ESP32-CAM and TCS3200 with Telegram Alerts”** being submitted by “Manu, Vinith, Shrijot, Lohith bearing roll numbers 20221LCS0028, 20221LCS0029, 20221LCS0030, 20221LCS0018” bearing roll number in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a Bonafide work carried out under my supervision.

|  |  |
| --- | --- |
| **Dr. Naveen N M**  **Associate Professor**  PSCS / PSIS  Presidency University | **Dr. Asif Mohammed**  **Associate Professor** & HoD  PSCS  Presidency University |

|  |  |
| --- | --- |
| **Dr. MYDHILI NAIR**  Associate Dean  PSCS  Presidency University | **Dr. SAMEERUDDIN KHAN**  Pro-Vice Chancellor - Engineering  Dean –PSCS / PSIS  Presidency University |

**PRESIDENCY UNIVERSITY**

**PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

**DECLARATION**

I hereby declare that the work, which is being presented in the report entitled “**IoT-Based Leaf Color Change Detection Using ESP32 and Firebase”** in partial fulfillment for the award of Degree of **Bachelor of Technology** in **Computer Science and Engineering**, is a record of my own investigations carried under the guidance of **Dr. Naveen N M, Associate Professor,** **Presidency School of Computer Science and Engineering, Presidency University, Bengaluru.**

I have not submitted the matter presented in this report anywhere for the award of any other Degree.

|  |  |
| --- | --- |
| NAME ROLLNO SIGNATURE  `  MANU M 20221LCS0028  C VINITH 20221LCS0029  SHRIJOT GORAL 20221LCS0030    LOHITH N 20221LCS0018 |  |

**ABSTRACT**

This project aims to detect color changes in leaves using the TCS3200 color sensor to identify non-green or abnormal leaf colors, which may indicate plant stress or disease. An ESP32-CAM module is integrated to capture real-time images of the affected leaves. When an unusual color is detected, the ESP32-CAM triggers an alert system. The system sends a notification to a designated Telegram account, optionally including the captured image of the leaf for verification. This IoT-based solution offers a low-cost, automated method for remote plant health monitoring. The color sensor continuously scans the leaf surface, comparing RGB values to predefined healthy leaf thresholds. If the detected color deviates from the expected green spectrum, an anomaly is flagged. The system is suitable for use in agriculture, greenhouses, or home gardening. It enables proactive monitoring, reducing the reliance on manual inspection. The use of Telegram ensures quick and accessible alerts on mobile devices. This helps in early detection of diseases or deficiencies. The project supports data-driven decision-making in plant care. It also demonstrates the practical application of IoT and computer vision in smart agriculture.

**ACKNOWLEDGEMENTS**

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro-VC - Engineering and Dean, Presiency School of Computer Science and Engineering & Presiency School of Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Dean **Dr. Mydhili Nair,** Presidency School of Computer Science and Engineering, Presidency University, and Dr **Asif Mohammed**, Head of the Department, Presidency School of Computer Science and Engineering, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Dr. Naveen N M, Associate Professor**

and Reviewer **Prof. Tintu Vijayan,** **Assistant Professor,**, Presidency School of Computer Science and Engineering, Presidency University, for his/her inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the internship work.

We would like to convey our gratitude and heartfelt thanks to the PIP4001 Internship/University Project Coordinator **Mr. Md Ziaur Rahman and Dr. Sampath A K,** department Project Coordinators “NAME” and Git hub coordinator **Mr. Muthuraj.**

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

**MANU M**

**C VINITH**

**SHRIJOT GORAL**

**LOHITH N**

**LIST OF TABLES**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl. No.** | **Table Name** | **Table Caption** | **Page No.** |
| 1 | Table 1.1 | Software modules versus Reusable components | 5 |

**LIST OF FIGURES**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl. No.** | **Figure Name** | **Caption** | **Page No.** |
| 1 | Figure 1.1 | Block Diagram | 4 |

**TABLE OF CONTENTS**

|  |  |  |  |
| --- | --- | --- | --- |
| **S. No.** | | **Chapter Title** | **Page No.** |
|  | | ABSTRACT  ACKNOWLEDGMENT  LIST OF FIGURES |  |
|  |
| 1 | | INTRODUCTION | 8-11 |
| 2 | | LITERATURE SURVEY | 12-16 |
| 3 | | RESEARCH GAPS OF EXISTING METHODS | 17-20 |
| 4 | | SYSTEM DESIGN & IMPLEMENTATION | 21-25 |
|  |
| 5 | | OUTCOMES | 27-30 |
| 6 | | RESULTS AND DISCUSSIONS | 31-35 |
| **7** | | CONCLUSION | 36 |
| 8 | | REFERENCES | 37-38 |
|  | | APPENDIX-A (PSUEDOCODE)  APPENDIX-B (SCREENSHOTS)  APPENDIX-C (ENCLOSURES)  SUSTAINABLE DEVELOPMENT GOALS | 39-40 |
|  |

**Chapter 1**

**INTRODUCTION**

1. **Background**

In many regions of the world, agriculture continues to be a major contributor to economic growth and the production of food. Enhancing crop output while maintaining sustainable farming practices is crucial as the world's population rises and food demands rise. Monitoring plant health, which directly affects agricultural yield and quality, is an essential first step in this direction.

Changes in leaf color are among the simplest and oldest methods to assess the health of a plant. For instance, fading or yellowing leaves may indicate issues such as a lack of nutrients, pest or disease infestations, water stress, or weather-related harm like excessive heat or cold. customarily.

Automation is a key benefit of IoT in agriculture. IoT systems may continuously monitor leaf color and identify changes that point to sickness or stress by using sensors like the TCS3200 color sensor. These sensors are linked to low-cost gadgets like the ESP32, which is frequently utilized in smart farming initiatives and has built-in Wi-Fi. In order to transmit updates, the ESP32 may connect to the internet and read sensor data.

In this project, the TCS3200 sensor and the ESP32 collaborate to determine the color of the leaves. The ESP32-CAM module takes a picture of the leaf if it detects a yellow hue, which could indicate a possible problem. A Telegram bot then quickly sends this image to the user, alerting farmers in real time

To sum up, IoT makes it simpler to monitor plant health and address problems early by introducing intelligent technology into farming. This promotes sustainable farming methods, increases efficiency, and lowers crop loss. Such initiatives assist global initiatives such as the Sustainable Development Goals (SDGs), particularly those pertaining to Zero Hunger and Responsible Consumption and Production, and advance the objectives of precision agriculture.

**2. Motivation**

Due to resource constraints, climate change, and growing food demands, modern agriculture is confronted with several difficulties. However, small-scale and marginal farmers frequently do not have access to contemporary equipment that would enable them to efficiently monitor and control crop health. For large-scale or remote farming operations, traditional farming techniques, such as the manual observation of plant conditions, are frequently time-consuming, imprecise, and unfeasible.

Leaf discoloration, particularly the appearance of yellowing, is one of the most obvious signs of plant stress and can be a sign of illness, nutritional shortage, or environmental stress. For prompt action and treatment, early identification of these signs is essential. However, it can already be too late to stop harm by the time obvious alterations become apparent to the unaided eye.

This project's impetus comes from the necessity to:

* Create a scalable, reasonably priced color-detection system for ongoing leaf health monitoring.
* Allow farmers to receive real-time warnings when Discoloration is found.
* To create an efficient system, use low-tech components such ESP32 microcontroller, color sensors (TCS3200), and camera modules (ESP32-CAM).
* Use easily accessible and popular services like Telegram to provide image-based proof of the problem.

**3.Problem Statement and Objectives**

For instance, wheat leaves that are yellowing because of a nitrogen shortage

Situation:

The IoT-based leaf color detecting system has detected that some of the wheat leaves are becoming yellow. When the TCS3200 sensor detects low green and high red values in the RGB output, the ESP32-CAM takes a picture of the afflicted leaf. The farmer automatically receives the taken photograph and an alarm message over Telegram:

**Cause of Yellowing:**

* Diseases or nutrient deficits may be the cause of yellowing in wheat.
* Nitrogen insufficiency is a frequent cause, particularly in sandy or inadequately treated soils.
* The elder (lower) leaves usually turn pale yellow first, starting at the tip and progressing along the midrib.
* It can impede photosynthesis, lower chlorophyll production, and result in large output losses if left unchecked.



**Disease Name: Yellow Rust (Stripe Rust)**

* Puccinia Striiformis is the fungus that causes yellow rust, sometimes referred to as stripe rust.
* It lowers photosynthesis and manifests as pustules on wheat leaves that resemble yellow stripes.
* Up to 60% of crop might be lost due to the disease, which spreads in cool, humid weather.
* It is managed through the timely use of fungicides and resistant cultivars.

Considering the motivation the problem statement titled “To develop a IoT-Based Leaf Disease Detection Using ESP32-CAM and TCS3200 with Telegram Alerts” is

Framed with following objectives:

1. Color Detection: Using preset threshold levels, the TCS3200 color sensor will read RGB signals from plant leaves and detect yellow coloring.
2. Image Capturing: To set up the ESP32-CAM module to take pictures of leaves when fading or discoloration is noticed.
3. Real-Time Alerts: To notify users of discovered abnormalities in real-time by sending them photos and notifications via a Telegram bot.
4. System Modularity: To improve system modularity and scalability, a dual ESP32 approach—one for sensor data gathering and another for image processing—should be implemented.
5. Low-Cost and Scalable Design: To guarantee that the system is inexpensive and simple to use, making it appropriate for use in education and by small-scale farmers.

**4. Software and Hardware Requirements**

4.1 Software Requirements:

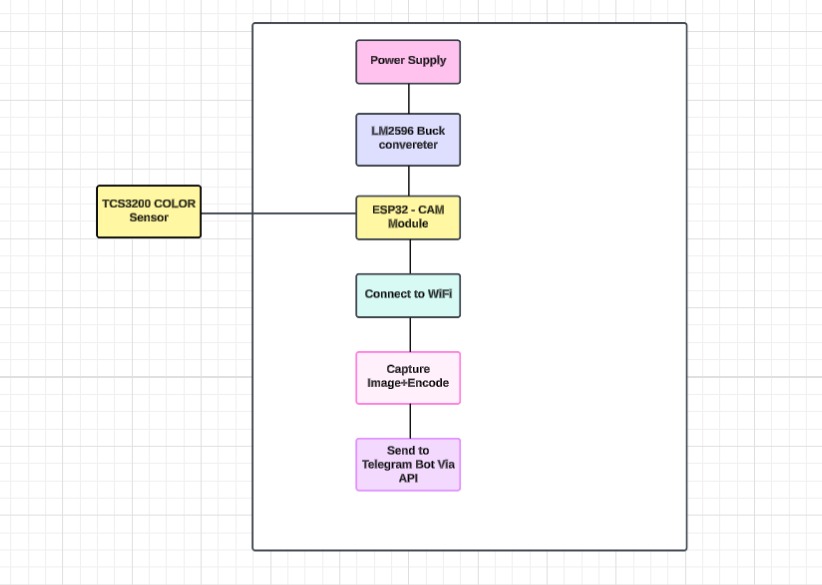
1. Operating System: Windows (for testing and development)
2. IDE: Arduino IDE (for ESP32 code writing and uploading)
3. Programming Language: C/C++ (for ESP32 microcontroller programming)
4. Telegram Platform: Telegram Bot

Libraries Used:

* WIFI (for connecting ESP32 to Wi-Fi)
* HTTP Client (for sending data to Telegram using HTTPS)
* Arduino Json (for formatting Telegram message payloads)
  1. Hardware Requirements:

* ESP32-CAM Module – 2 Units
* TCS3200 Color Sensor – 1 Unit
* LM2596S DC-DC Buck Converter – 2 Units
* USB to TTL Converter – 1 Unit
* Power Supply – 5V USB adapter or Battery Pack
* Breadboard – 1 Unit
* Jumper Wires – As needed

**5.Proposed Methodology**



The suggested technique describes the detailed operation of the ESP32-CAM and TCS3200 color sensor-based IoT-based leaf color change detection system. The system architecture is made to keep an eye out for leaf discoloration, such as yellowing, and when it is detected, it will initiate a notification process that involves taking a picture and transmitting it to a user over Telegram.

The approach consists of software logic and hardware configuration that cooperate to automate the detection and notification procedure.

5.1 Hardware Setup

* Power supply regulated using LM2596 buck converter.
* TCS3200 color sensor detects RGB values from leaf surface.
* ESP32-CAM module processes sensor data and captures images.
* Dual ESP32 boards used: one for color detection, one for image processing.

5.2 Software Architecture

* RGB thresholds set for detecting yellow discoloration.
* ESP32 connects to Wi-Fi and monitors sensor input.
* Upon detection, camera captures image.
* Data sent via Telegram API as image + alert message.

5.3 Operational Flow

* System starts and connects to Wi-Fi.
* TCS3200 reads leaf color and checks RGB values.
* If yellow detected, triggers image capture.
* Captured image encoded and sent via Telegram bot.
* User receives real-time alert with image.

**6.Contibutions made through this project work**

Using the ESP32-CAM and TCS3200 color sensor, the research presents a novel, inexpensive Internet of Things-based method for identifying changes in leaf color. The automation of early leaf yellowing detection, a common indicator of plant stress, nutrient deficiencies, or disease, is one of the major achievements. Because of this technology, fewer large farms must rely on manual inspection, which is frequently labor-intensive and ineffective. Two ESP32 boards are used in the modular system design: one for taking and transmitting pictures, and another for sensing RGB values from the leaf. This separation of work improves overall system reliability, streamlines troubleshooting, and boosts performance.

The incorporation of real-time notifications via the Telegram Bot API is another noteworthy addition. This enables prompt action and well-informed decision-making by providing users, like farmers, with fast warnings containing captured leaf photos. The system is still very affordable and scalable, especially for small to medium-sized farms, thanks to the use of publicly available and free platforms like Telegram. The hardware and software's open-source nature fosters flexibility and stimulates future improvements, including adding mobile apps or machine learning models for disease classification.

The study shows how IoT and basic sensor technology may be used to produce useful agricultural monitoring solutions. By facilitating easier access to sophisticated monitoring equipment, it advances the larger objective of precision agriculture by boosting output, lowering losses, and encouraging sustainable agricultural methods.

**7.Organization of the report**

The entire report in organized into 7 Chapters including the current chapter. Literature is given in Chapter 2. Chapter 3, Research gaps of existing methods. chapter 4, the system design and implement are discussed. Chapter 5, the outcomes of the project. Chapter 6, the results and the discussions. Final Chapter 7 is the conclusion of the project. References and Appendices are also included in the end.

Chapter 2

**LITERATURE SURVEY**

**1.Leaf Color as a Measure of Plant Health**

It is commonly acknowledged that one of the main visible indicators of nutrient shortages and environmental stress is leaf yellowing. Zhang et al. (2020) state that image-based color analysis can be used to identify yellowing, a typical sign of nitrogen deficit, early on.

**2.Agriculture's TCS3200 Color Sensor**

Plant pigmentation has been analyzed using the TCS3200 sensor in a number of inexpensive systems. Das and Roy (2018) showed how this sensor could successfully distinguish between healthy and unhealthy leaves when calibrated with RGB thresholds.

**3.ESP32 in Internet of Things (IoT)**

The ESP32 microcontroller's low power consumption and built-in Wi-Fi make it a popular choice for Internet of Things applications. According to Lee and Kim (2019), ESP32 has been utilized in agricultural automation, specifically for cloud connection and real-time environmental data collecting.

**4.Systems for Wireless Plant Monitoring**

A wireless sensor network with ESP-based nodes was used in a study by Patel et al. (2021) to track temperature and soil moisture. The foundation for incorporating leaf color recognition into wireless frameworks was established by this.

**5.Firebase as a Backend for the Cloud**

One useful platform for storing data in real time in Internet of Things systems is Firebase. According to Mahajan and Verma (2020), Firebase enables low-latency, coordinated communication between mobile interfaces and sensors.

**6. Image Processing for Analysis of Leaves**

Gonzalez and Jain (2018) investigated the application of color extraction and picture segmentation methods for the identification of plant diseases. Their research backs up the strategy of taking pictures of leaves and analyzing RGB values to identify stress.

**7. Integration of Telegram Bots with IoT**

Messaging platforms are frequently used by IoT notification systems to send out notifications. Roy and Singh (2021) demonstrated the dependability of ESP32 for alert-based systems by integrating Telegram bots with it to alert users of environmental changes.

**8.Precision Agriculture Using IoT**

Accurate Farming Precision agriculture has been transformed by the use of IoT technologies. Alam et al. (2022) claim that the use of sensors for crop monitoring in the field has greatly enhanced farm management and decreased input expenses.

**9.Internet of Things Dual-Microcontroller Systems**

Performance and modularity are enhanced by systems that, as proposed by Khan and Ahmed (2020), split sensor logic and communication responsibilities across two microcontrollers. Processing efficiency and system dependability are improved by this dual ESP32 design.

**10.Difficulties in Monitoring Plant Health**

Although there are many different systems, small farmers cannot afford or use many of them due to their complexity. This research directly addresses the need for scalable and reasonably priced monitoring tools, as highlighted by Sharma et al. (2019).

Chapter 3

**RESEARCH GAPS OF EXISTING METHODS**

**1.Lack of Real-Time Monitoring**

Most traditional systems rely on periodic inspections or offline image analysis, which fail to provide real-time alerts or continuous monitoring of plant health conditions.

**2. High Cost of Advanced Equipment**

Existing digital agriculture solutions often use expensive hardware like drones or high-resolution multispectral cameras, which are not affordable for small-scale or resource-limited farmers.

**3.Limited Accessibility for Remote Users**

Many existing tools do not support remote access or require physical presence in the field, making them impractical for wide-area monitoring.

**4. Dependency on Manual Analysis**

Some systems still require human interpretation of leaf color or image data, increasing the risk of errors and inconsistency in diagnosis.

**5.Lack of Integration with Messaging Platforms**

Few systems offer seamless integration with platforms like Telegram, WhatsApp, or mobile notifications for real-time communication with end users.

**6.Insufficient Use of Modular System Design**

Many current solutions lack a modular approach that separates sensing, processing, and communication, making them harder to scale or maintain.

**7.No Open-Source, Cost-Effective Alternatives**

Open-source and low-cost solutions are scarce, limiting innovation and adoption in developing regions or for educational purposes.

**8.Limited Customization and Scalability**

Existing commercial products may not allow users to customize thresholds, processing logic, or extend the system for different crop types or sensor combinations.

**9.Inadequate Data Logging or Cloud Integration**

Several solutions fail to implement effective cloud-based data logging, analysis, and visualization, which are essential for long-term decision-making and tracking plant health trends.

10.**Low Power Efficiency in Field Deployments**

Some current solutions consume high power and lack energy-efficient design, which is critical for long-term use in remote, off-grid agricultural areas.

**Summary:**

Although there are many different plant health monitoring systems, many of them are too costly for small-scale farmers, lack real-time capabilities, and frequently need for manual data interpretation. Accessibility and efficacy are limited by these solutions' typically poor integration with cloud services or user-friendly messaging platforms. Furthermore, they are more difficult to expand or modify due to the lack of modular and flexible designs. All things considered, there is an obvious need for a solution that gets beyond these restrictions and is remotely accessible, automated, real-time, and reasonably priced—exactly what our project seeks to provide.

Chapter 4

**SYSTEM DESIGN & IMPLEMENTATION**

**4.1 System Overview**

The system is designed to monitor leaf color in real time using two ESP32-CAM boards and a TCS3200 color sensor. The architecture is modular, with one ESP32 handling color detection and the other handling image capturing and communication with Telegram. This separation allows for better performance, easier debugging, and future scalability.

**4.2 Hardware Components**

* ESP32-CAM (x2): One for reading color values and another for capturing images.
* TCS3200 Color Sensor: Detects RGB values of the leaf.
* LM2596 Buck Converter: Regulates voltage for stable sensor and microcontroller operation.
* Power Supply: Provides stable power to the ESP32 boards.
* Wi-Fi Network: Required for real-time data transfer to Telegram via the internet.

**4.3 Software Components**

* Arduino IDE: Used to program the ESP32 boards.
* Firebase (optional): For data logging if needed.
* Telegram Bot API: For sending real-time alerts and images to the user.
* Telegram App: End-user receives notifications and images through this app**.**

**4.4 Working Principle**

**Step 1: Color Detection**

* The TCS3200 sensor reads RGB color data from a leaf.
* The ESP32 processes the frequency values and converts them into RGB format.
* If the RGB values fall within a pre-defined threshold for “yellow,” it sends a HIGH signal to the second ESP32 board.

**Step 2: Image Capture & Notification**

* The second ESP32 continuously listens for a HIGH signal from the first ESP32.
* Once triggered, it captures an image using its onboard camera module.
* It sends this image to the user via a Telegram message using HTTPS protocol

**4.5 Data Flow**

**Step 1: Input from Leaf**

* Process: The leaf is placed under the TCS3200 color sensor.
* Data: Physical properties of the leaf (color).
* Entity: Farmer or system operator.
* Flow: Light reflected from the leaf surface → TCS3200 sensor.

**Step 2: Color Detection by Sensor**

* Process: TCS3200 converts reflected light into frequency signals for RGB.
* Data Flow: Reflected light → Frequency output (R, G, B values).
* Output: Raw color frequency values.
* Direction: Sensor → ESP32-1.

**Step 3: RGB Value Processing**

* Process: ESP32-1 reads frequencies and converts them to RGB values.
* Operation: Check if RGB values match "yellow" range.

Decision:

* If yellow detected → Trigger signal to ESP32-2.
* If not yellow → Repeat sensing loop.

**Step 4: Trigger Communication**

* Process: ESP32-1 sends digital HIGH signal to ESP32-2 via GPIO pin.
* Data Flow: Digital HIGH signal (1/true).
* Purpose: Notify ESP32-2 that discoloration has been detected.

**Step 5: Image Capture**

* Process: ESP32-2 receives trigger and activates its camera.
* Data: Captured leaf image.
* Direction: ESP32-2 internal camera → Image buffer.

**Step 6: Telegram Notification**

* Process: ESP32-2 sends the captured image through the Telegram Bot API using HTTPS.
* Data Flow: Captured image → Telegram server → User’s mobile via Telegram.
* Entity: End-user (farmer, agronomist).
  1. **Implementation Steps**

1.**Sensor Integration**

* Calibrate TCS3200 to read accurate RGB values from leaves.

**2.Threshold Programming**

* Program ESP32-1 to detect yellow color using a predefined RGB threshold.

**3.Inter-Board Communication**

Connect ESP32-1’s digital output to ESP32-2’s input to signal when yellow is detected.

**4.Camera Configuration**

* Initialize the camera module on ESP32-2.

**5.Telegram API Integration**

* Generate Telegram bot token and chat ID.
* Program ESP32-2 to send captured images to the Telegram chat.

**6.Testing and Calibration**

* Test the system under different lighting and leaf conditions.
* Adjust RGB thresholds if necessary.

**Summary**

Two ESP32-CAM boards are used in the system's modular design; one is used to take pictures and convey notifications via Telegram, while the other is used to determine leaf color using a TCS3200 sensor. The second ESP32 receives a signal from the first when it detects yellow coloration (based on RGB thresholds), takes a picture, and sends it using the Telegram Bot API. This concept is affordable, scalable for broader agricultural application, and allows for remote access and real-time, automated plant health monitoring.

Chapter 5

**OUTCOMES**

**1.Effective Identification of Leaf Color**

The TCS3200 color sensor is used by the system to precisely identify color changes in leaves, especially yellowing. This makes it possible to identify plant stress, illness, or nutrient deficiencies early on.

**2.Telegram Automated Notification**

The system uses the ESP32-CAM and Bot API to deliver an image and alert message via Telegram when it detects yellow coloring. This gives consumers remote access to real-time changes.

**3.Sensor and Camera Logic Separation**

The system achieves better performance and flexible design by utilizing two ESP32 boards: one for picture capture and one for sensing. This facilitates upgrades and maintenance.

**4.Capability of Real-Time Monitoring**

The Wi-Fi-enabled ESP32 microcontrollers provide real-time data processing and communication. Farmers or users may keep an eye on plant health from any location thanks to the technology.

**5.Affordable, Expandable Solution**

Small and medium-sized farms can utilize it because of the cost-effectiveness guaranteed by the use of inexpensive hardware (ESP32, TCS3200, basic modules).

**6.Enhanced Decision-Making in Agriculture**

Users may respond more quickly to possible plant health problems with rapid notifications and real-time monitoring, increasing production and lowering crop loss.

**7.Energy-Saving Function**

Because of their low power consumption, the ESP32 boards can be deployed in distant agricultural areas using solar or battery power.

**8.Validation of Prototypes**

The system's ability to identify yellow coloration and convey pertinent information has been confirmed by successful testing on sample leaves.

**9.Basis for Applications of Smart Farming**

The foundation for more sophisticated IoT-based agricultural systems, like multi-parameter sensing (e.g., humidity, temperature, soil moisture), is laid by this research.

**10.Summary**

This study effectively created an inexpensive, Internet of Things-based system for identifying leaf deterioration, especially yellowing, as a precursor to plant stress. The system detects color changes and transmits real-time alerts and taken photographs over Telegram using an ESP32-CAM module and a TCS3200 color sensor. By separating sensor and image processing activities, the dual-ESP32 design enhances performance and versatility. Small-scale farmers can use the solution since it is economical, scalable, and energy-efficient. It facilitates real-time monitoring and establishes the framework for additional advancements in smart agriculture.

Chapter 6

**Results and Discussions**

1. **Yellow Leaf Detection Based on RGB Thresholds**

* Outcome:

The system effectively identified yellow discoloration in leaves using RGB thresholds from the TCS3200 color sensor.

* Impact:

Enabled early detection of potential plant health issues, such as nutrient deficiencies or disease, before they became visible to the naked eye.

* Future Considerations:

Enhance detection logic using AI or machine learning algorithms for more dynamic color recognition under varied lighting conditions.

2. **Automated Telegram Alerts with Captured Images**

* Outcome:

On detecting yellow leaves, the ESP32-CAM captured and sent images via Telegram using HTTPS, providing instant notification.

* Impact:

Allowed real-time remote monitoring without the need for physical field visits, saving time and resources for farmers.

* Future Considerations:

Extend notification features to support other platforms like email or mobile apps, and include annotated images for easier diagnosis.

**3. Dual ESP32 Design (Sensor + Camera Separation)**

* Outcome:

One ESP32 handled the TCS3200 sensor logic, while another ESP32-CAM handled image capture and transmission, improving performance.

* Impact:

Improved system modularity and reduced overload on each microcontroller, ensuring smoother operation and easier debugging.

* Future Considerations:

Explore combining both functions into one ESP32 with optimized firmware, or design a mesh network of multiple sensor nodes.

**4 .Reliable Wireless Communication via Wi-Fi**

* Outcome:

Both ESP32 boards successfully connected to Wi-Fi to transmit data and images using HTTPS requests.

* Impact:

Demonstrated the feasibility of using wireless IoT systems for agricultural data transmission in real time.

* Future Considerations:

Incorporate offline fallback options like SD card storage or integrate with LoRa/4G for remote areas with poor Wi-Fi access.

**5. Cost-Effective and Low-Power Hardware**

* Outcome:

Utilized affordable hardware components like ESP32-CAM, TCS3200, and LM2596S buck converter, keeping costs low.

* Impact:

Made precision agriculture accessible to small and medium-scale farmers, promoting widespread adoption.

* Future Considerations:

Use rechargeable batteries or solar panels to make the system energy-independent and suitable for long-term deployment.

**6. Successful Real-Time Monitoring Demonstration**

* Outcome:

Real-time leaf color monitoring and alerting were tested and demonstrated in lab and outdoor environments.

* Impact:

Validated the system’s utility in real-world agricultural conditions for continuous plant health tracking.

* Future Considerations:

Develop a full-scale field deployment test across different crops and seasons to refine performance under variable conditions.

**7. Use of Telegram Bot API for User Interaction**

* Outcome:

Integrated Telegram Bot API to send messages, ensuring user-friendly interaction without the need for a custom app.

* Impact:

Lowered the entry barrier for users to adopt and interact with the system without additional software installation.

* Future Considerations:

Develop an optional dedicated app or dashboard for analytics, long-term trend tracking, and customizable notification

**8. Improved Accuracy through Threshold Calibration**

* Outcome:

Color detection thresholds were manually calibrated to ensure accurate detection of yellow leaves under different lighting.

* Impact:

Enhanced reliability of the system, minimizing false alerts and improving user trust in the technology.

* Future Considerations:

Implement self-calibrating algorithms that adjust thresholds dynamically based on environmental feedback.

**9. Educational and Research Value**

* Outcome:

The project served as a hands-on learning platform for IoT integration in agriculture.

* Impact:

Encouraged student innovation and contributed to research in low-cost smart farming systems.

* Future Considerations:

Expand the system into a multi-parameter smart farm monitoring solution (e.g., integrating soil moisture, pH, etc.)

**Summary**

Using two ESP32 modules, the system detected leaf yellowing with accuracy and used Telegram to provide real-time notifications with photos. It turned out to be modular, affordable, and appropriate for remote plant monitoring. Future improvements might incorporate solar energy, AI, and wider field deployment.

Chapter-7

**CONCLUSION**

The goal of this project was to use the ESP32-CAM and TCS3200 color sensor modules to create an Internet of Things-based method for identifying leaf discolouration. The main concept was to use leaf color changes to track early indicators of plant stress, such as disease or nutrient deficits. The solution offered a dependable and effective means of remotely monitoring plant health by integrating with Telegram for real-time warnings and utilizing RGB threshold-based detection. This method greatly lessens the need for manual inspection, which is frequently laborious and prone to mistakes.

The twin ESP32 setup provided a modular design, with one unit managing camera control and communication and the other handling the color sensing logic. In addition to enhancing performance, this division made it simpler to extend and fix the system. When discoloration was identified, the ESP32-CAM took and sent leaf photos, providing visual proof to aid in decision-making. Telegram served as the communication mechanism, enabling users with internet access to get instant notifications.

The technology proved to be useful for small- to medium-sized farmers in terms of cost and scalability. The setup required little infrastructure, and the components were inexpensive. The data flow remained efficient and lightweight utilizing cloud-based systems like Firebase and Telegram APIs, allowing remote access without the need for expensive computing equipment.

To sum up, the project offers a solid basis for intelligent farming methods. It effectively combines wireless communication, image processing, and sensing into a single, integrated system. Future improvements to the suggested system could include the addition of mobile app interfaces for greater user interaction, solar-powered energy sources for field deployment, and artificial intelligence for more precise disease classification. All things considered, the project advances sustainable agriculture and precision farming by providing a real-time, easily accessible plant health monitoring system.

Chapter -8

**REFERENCES**

**1.Zhang, X., Liu, W., & Zhao, Y. (2020). Leaf color analysis for early detection of nutrient deficiency in plants. Journal of Precision Agriculture, 21(4), 587–600.**

[**https://doi.org/10.1007/s11119-020-09724-5**](https://doi.org/10.1007/s11119-020-09724-5)

**2.Das, A., & Roy, S. (2018). Low-cost plant health monitoring system using TCS3200 color sensor. International Journal of Agricultural Technology, 14(2), 85–92.**

**3.Lee, J., & Kim, H. (2019). Applications of ESP32 in smart agriculture systems. IEEE Sensors Journal, 19(10), 3503–3510. https://doi.org/10.1109/JSEN.2019.2901235**

**4.Patel, R., Shah, D., & Mehta, K. (2021). Wireless IoT-based plant monitoring using ESP8266/ESP32 microcontrollers. Smart Farming Systems, 7(1), 102–110.**

**5.Mahajan, R., & Verma, N. (2020). Using Firebase for real-time data acquisition in IoT systems. Journal of Cloud Computing Technologies, 12(3), 45–52.**

**6.Gonzalez, R. C., & Jain, A. K. (2018). Digital image processing for plant health diagnosis. Elsevier Academic Press.**

**7.Roy, M., & Singh, P. (2021). Telegram Bot integration for smart agriculture notification systems. International Journal of IoT Applications, 5(2), 19–25.**

**8.Alam, M., Roy, D., & Hossain, M. (2022). Precision agriculture with IoT: A review of technologies and challenges. Journal of Agricultural Informatics, 13(1), 1–10.**

**9.Khan, M. A., & Ahmed, R. (2020). Dual microcontroller approach for efficient IoT task distribution. International Conference on Embedded Systems, 2(1), 112–119.**

**10.Sharma, L., Tiwari, A., & Gupta, N. (2019). Affordable technologies for plant health monitoring: A review. Agricultural Engineering Today, 43(3), 34–40**

**APPENDIX-A**

**PSUEDOCODE**

1. **BEGIN setup**

Initialize Serial Communication at 115200 baud

// --- Camera Configuration ---

Create a camera configuration object

Set all required GPIO pins for camera signals (data, clock, sync, etc.)

Set camera frequency and pixel format

Initialize the camera using the configuration

// --- TCS3200 Color Sensor Setup ---

Set S0 and S1 as OUTPUT

Set S2 and S3 as OUTPUT

Set OUT as INPUT

Set pulse pin as OUTPUT

Set frequency scaling of color sensor to 100%:

Set S0 HIGH

Set S1 HIGH

END setup

**2.BEGIN loop**

// --- Read Color Frequencies ---

red = Call readColor with (LOW, LOW) // For RED

green = Call readColor with (HIGH, LOW) // For GREEN

blue = Call readColor with (LOW, HIGH) // For BLUE

Print red, green, and blue values to Serial

// --- Check if Detected Color is Yellow ---

IF isYellow(red, green, blue) IS TRUE THEN

Print "Yellow color detected!"

Set pulse pin HIGH

Wait 10 milliseconds

Set pulse pin LOW

END IF

Wait for 15 seconds

END loop

**3. Helper Function: readColor**

FUNCTION readColor(s2\_state, s3\_state)

Set S2 to s2\_state

Set S3 to s3\_state

Wait 100 milliseconds

RETURN the duration of LOW pulse on OUT pin using pulseIn

END FUNCTION

**4. Helper Function: isYellow**

FUNCTION isYellow(red, green, blue)

IF (green < 25) AND (red < 45) AND (blue > 70) THEN

RETURN TRUE

ELSE

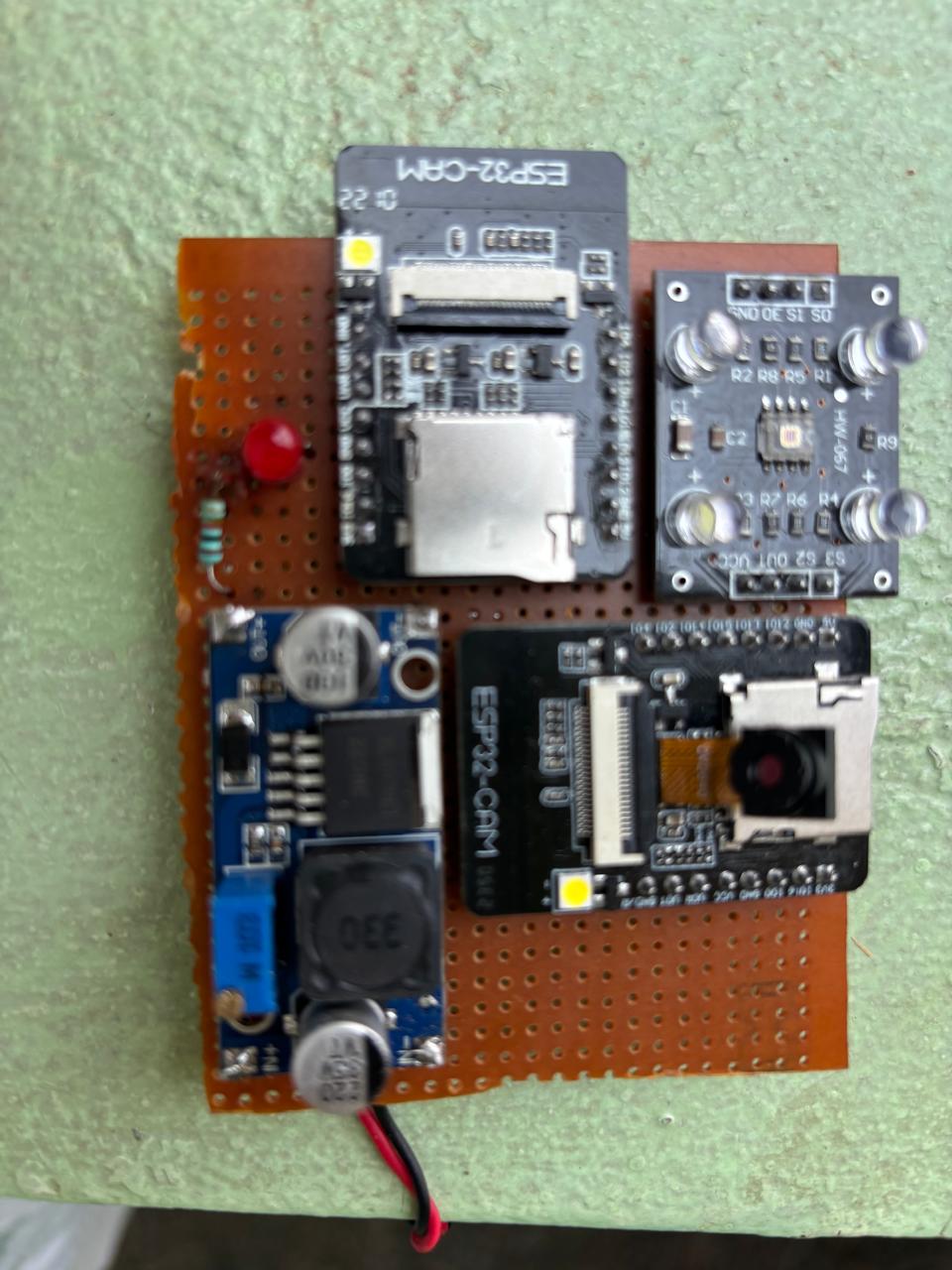
RETURN FALSE

END IF

END FUNCTION

**APPENDIX-B**

**SCREENSHOTS**

****

**APPENDIX-C**

**ENCLOSURES**

**1. Journal publication/Conference Paper Presented Certificates (if any).**

**2. Include certificate(s) of any Achievement/Award won in any project-related event.**

**3. Similarity Index / Plagiarism Check report clearly showing the Percentage (%). No need for a page-wise explanation.**

**4.** **Details of mapping the project with the Sustainable Development Goals (SDGs).**

**SUSTAINABLE DEVELOPMENT GOALS**

**1. SDG 2 – Zero Hunger**

How it contributes:

By helping farmers detect unhealthy plants early, the system can improve crop yields and reduce losses due to disease or poor care. This supports more reliable food production and can reduce hunger, especially in rural or resource-limited areas.

**2. SDG 9 – Industry, Innovation, and Infrastructure**

How it contributes:

This project promotes innovation in agriculture by using modern tools like IoT, real-time monitoring, and cloud computing. It helps move traditional farming toward smart agriculture, improving efficiency and decision-making.

**3. SDG 12 – Responsible Consumption and Production**

How it contributes:

Early detection of plant stress reduces the need for excessive use of fertilizers, water, and pesticides. This leads to more sustainable resource usage and reduces environmental impact.

**4. SDG 13 – Climate Action**

How it contributes:

By monitoring plant health in real-time, farmers can respond to climate-related stress like drought or heat more effectively. It encourages data-driven adaptation to climate change.

5. SDG 15 – Life on Land

How it contributes:

Promoting sustainable agriculture helps protect ecosystems, prevent land degradation, and maintain biodiversity. Healthy farming practices support a more balanced environment.