A

MINI PROJECT REPORT ON

ROBOTIC ARM USING IMAGE PROCESSING

Submitted by

Mr.Mayur Ichake **Exam Seat No.T400600197**
Mr.Sahil Ichake **Exam Seat No. T400600198**
Mr.Aditya Mungase **Exam Seat No. T400600160**

Under the guidance of
Prof. Dhananjay Poul

In the fulfillment of T.E. in Electronics and Telecommunications Engineering course of Savitribai Phule Pune University in academic Year 2024-25

Department of Electronics & Telecommunication Engineering
Dhole Patil College of Engineering, Pune

Savitribai Phule Pune University

2024-2025

DHOLE PATIL EDUCATION SOCIETY's

DHOLE PATIL COLLEGE OF ENGINEERING

Accredited by NAAC with A+ Grade, An ISO 9001:2015 Certified Institute

Near Eon IT Park Kharadi, Dhole Patil College Road, Wagholi, Pune

Website www.dpcoepune.edu.in **E-mail:** dpcoepune@gmail.com,

CERTIFICATE

This is to certify that the Mini Project report entitled

'ROBOTIC ARM USING IMAGE PROCESSING'

submitted by

Mayur Ichake (Exam Seat No: T400600197),

Sahil Ichake (Exam Seat No: T400600198),

Aditya Mungase(Exam Seat No: T400600160)

is a record of bonafide work carried out by him/her under my guidance in the partial fulfillment and the requirement of the T.E. Electronics & Telecommunication Engineering, 2019 Course of Savitribai Phule Pune University, Pune in the academic year 2024- 2025.

Guide Name: Prof.Dhananjay Poul

Prof. Prachi Deshpande **External Examiner**

H.O.D.

Date: / / 2025 **Prof. Dr.Abhijit Dandavate**

Place: Wagholi, Dist.: Pune. **Principal**

ACKNOWLEDGEMENT

I express my sincere gratitude to my project guide, **Prof.Dhananjay Poul**, Assistant Professor, Department of Electronics & Telecommunication Engineering, for her unwavering support, insightful guidance, and continuous encouragement throughout this project. Her expertise and valuable feedback were instrumental in shaping the outcome of this work.

I am also deeply thankful to **Prof.Prachi Deshpande,** Head of the Department of E&TC Engineering, for her valuable suggestions and support during this project. My heartfelt appreciation extends to **Dr.Abhijit Dandavate**, Principal of Dhole Patil College of Engineering, Pune, for providing the necessary support and resources to facilitate this work.

I would like to extend my special thanks to the Management of Dhole Patil College of Engineering, Pune, for offering the required infrastructure and facilities that contributed to the successful completion of this project.

I would like to thank all Teaching and Non-Teaching Staff members of the E&TC Engineering department and my friends for helping me directly or indirectly during the completion of my Project.

Contents

List of Figures

List of Abbreviations

ESP	Espressif system platform
CAM	Camera module
USB	Universal serial bus
ATmega	Advanced Technology megaAVR microcontroller
GND	Ground
VCC	Voltage Common Collector
LM	Linear Monolithic
PCA	Pulse Control Amplifier
Wi-Fi	Wireless Fidelity
DC	Direct Current

Abstract

Agriculture is a fundamental sector that is increasingly benefiting from advancements in automation and intelligent systems. This project focuses on the development of a robotic arm system enhanced with image processing capabilities for the automatic detection and harvesting of fruits or vegetables from trees or plants.

The goal is to reduce manual labor, improve efficiency, and ensure precision in the harvesting process. The system integrates a camera module (such as the OV2640) to capture real-time images of crops. These images are processed using computer vision techniques implemented through OpenCV to detect ripe produce based on predefined visual features such as color, shape, and size. The use of HSV color space helps in isolating the color of the targeted fruit for accurate segmentation and detection. Once a fruit is identified, the system calculates its coordinates within the image frame.This positional information is then communicated to a microcontroller unit—typically an Arduino or ESP32—which interprets the data and sends precise control signals to a robotic arm. The robotic arm, composed of multiple servo motors, is capable of multi-axis movement and is programmed to reach the detected location. A mechanical harvesting tool (such as a cutter or gripper) attached to the arm performs the cutting operation, ensuring minimal damage to the plant and the produce.The system is designed to operate in real-time and can be adapted to various crops by adjusting the detection parameters. This project highlights the effective combination of embedded systems, robotics, and computer vision to solve real-world agricultural problems. It serves as a prototype for future automated harvesting solutions that can help farmers increase productivity, save time, and reduce reliance on seasonal labor.

1. INTRODUCTION

1.1 Overview

This project presents a smart agricultural solution that combines image processing and robotics to automate the harvesting of fruits and vegetables. The system is designed to detect ripe produce on trees or plants using a camera and image recognition techniques, and then instruct a robotic arm to move to the detected location and harvest the item.The process begins with a camera module, such as the OV2640, capturing real-time images of the plant. These images are processed using computer vision techniques (such as color filtering and contour detection) to identify the presence and position of ripe fruits or vegetables. The image processing can be handled by a Raspberry Pi, ESP32-CAM, or similar processing unit.Once the fruit is located, its coordinates are sent to a microcontroller (Arduino or ESP32), which controls a robotic arm. The arm consists of multiple servo motors that allow movement in various directions. A cutting mechanism or gripper attached to the arm performs the harvesting action.This project demonstrates how embedded systems, robotics, and image processing can be integrated to create an efficient, low-cost system for precision agriculture. It has the potential to reduce manual labor, increase harvesting efficiency, and be adapted to different types of crops with minimal changes to the software.

1.2 Necessity

In today's agricultural practices, a significant amount of time and labor is spent on harvesting crops, especially fruits and vegetables that require careful handling. Manual harvesting is not only labor-intensive but also inconsistent, costly, and prone to human error or damage to produce. With the growing demand for food and the shortage of skilled labor in the farming sector, there is a strong need for automation in agriculture. .

1.3 Objective

The objective of this project is to design and implement an automated system that can detect and harvest fruits or vegetables using image processing and a robotic arm. The system aims to reduce manual labor and improve the efficiency of agricultural practices. By using a camera module, the project focuses on detecting ripe produce based on features such as color, shape, or size. The image processing component calculates the position of the fruit and sends this information to a microcontroller, such as an Arduino or ESP32, which controls the movement of the robotic arm. The arm is designed to reach the fruit and perform the necessary cutting or picking action. Additionally, the project seeks to create a low-cost and efficient prototype that demonstrates how embedded systems, computer vision, and robotics can work together to solve real-world farming problems.

1.4 Problem Statement

In traditional farming, the process of harvesting fruits and vegetables is heavily dependent on manual labor, which is time-consuming, physically demanding, and often inefficient. This becomes a major challenge, especially during peak harvesting seasons or in regions where there is a shortage of skilled agricultural workers. Manual harvesting can also lead to inconsistent results, damage to the crops, and higher operational costs. Additionally, identifying ripe fruits accurately requires experience and constant human supervision. With the increasing demand for food production and the growing push for automation in agriculture, there is a need for a system that can detect and harvest crops in a precise, efficient, and automated manner. This project addresses the problem by developing a robotic arm system with image processing capabilities that can identify ripe produce and harvest it automatically, reducing labor dependency and improving overall productivity in agricultural operations.

1.5 Organization

This Work consists of five chapters. The following is the outline of this seminar.

Chapter 1 contains Introduction, necessity, objective, theme and organization of thesis.

Chapter 2 will discuss about research and information about the project. Every fact and information that found through journals or other references will be mentioned in this chapter.

Chapter 3 represents the methodology used. All methodology will be described details in this chapter.

Chapter 4 includes the performance analysis and software simulation.

Chapter 5 includes result and execution outputs.

Chapter 6 is the last chapter. This chapter will conclusion & future scope.

2. LITERATURE SURVEY

2.1. Overview and Purpose of Literature Survey

The concept of automating tasks in agriculture has evolved significantly over the years, with modern technologies such as robotics, image processing, and machine learning playing a key role in transforming farming practices. Traditionally, harvesting has been a manual process that requires a significant amount of labor, especially when it comes to fruit and vegetable crops. However, with increasing demand for efficiency and the shortage of skilled labor, there has been a drive towards developing automated systems capable of performing tasks like fruit detection and harvesting with minimal human intervention.Image processing has become a vital tool in detecting and identifying ripe produce. which are crucial for automated harvesting systems.In parallel, the integration of robotic arms into agriculture has revolutionized harvesting by providing precise control over movements. These robotic systems, equipped with grippers or cutting tools, use servo motors to move along multiple axes and harvest produce without damaging the plants. Research in agricultural robotics has focused on developing arms capable of handling a wide variety of fruits, from apples to citrus, ensuring high efficiency and precision.Microcontrollers such as Arduino and ESP32 are integral to the system's operation, processing inputs from sensors and cameras to control the robotic arm's movement. Motion sensors, including ultrasonic and infrared sensors, assist in distance measurement and help guide the arm to the fruit's exact location for harvesting. Combining these sensors with image processing ensures that the arm moves with accuracy and responds dynamically to its environment.Recent advancements in artificial intelligence and machine learning further enhance the system's capabilities. AI algorithms are used to improve the recognition of different types of fruits and to analyze the ripeness, which optimizes the timing of the harvest. These technologies also help in refining the movement of the robotic arm, ensuring the most efficient and energy-saving path to harvest the fruit.

2.2 Literature Review

1. Ege Gursoy, Benjamin Navarro, Akansel Cosgun, Dana Kulić, Andrea Cherubini , “Towards Vision-Based Dual Arm Robotic Fruit Harvesting ”, arXiv 2023
   Summary: This paper proposes a dual-arm robotic system for fruit harvesting, equipped with an RGB-D camera and cutting tools. The system employs a Hierarchical Quadratic Programming-based control strategy to handle constraints such as joint limits and potential collisions. The authors combine deep learning and standard image processing algorithms to detect and track fruits and tree trunks. The approach was validated through real-world experiments, showcasing the potential of dual-arm systems in complex harvesting tasks.

2. He, K., Zhang, X., Ren, S., Sun, J. , “Development of Tomato Detection Model for Robotic Platform Using Deep Learning ”, Multimedia Tools and Applications 2021
   Summary: his research focuses on developing a deep learning-based model for detecting tomatoes in agricultural environments. The study highlights the challenges of detecting fruits under varying lighting conditions and occlusions. The authors propose an optimized YOLOv3 algorithm tailored for tomato detection, demonstrating its effectiveness in real-time applications. The model's performance was evaluated in terms of accuracy and computational efficiency, making it suitable for deployment on robotic platforms.

3. J. R. Rosell-Polo, F. Andújar, A. Escolà, S. Martínez-Guanter, A. R. Camp and A. M. Llorens, "Advanced technologies for the improvement of spray application techniques in Spanish viticulture: An overview," *Computers and Electronics in Agriculture*, vol. 151, pp. 66–90, May 2018.
   Summary: This paper provides a comprehensive review of how advanced technologies are being applied to improve pesticide spray techniques in Spanish vineyards (viticulture). The goal is to optimize chemical usage, reduce environmental impact, and improve efficiency and effectiveness in disease and pest control.

2.3 Demerits of Existing System

Existing automated fruit harvesting systems, while offering significant improvements over manual labor, still have several drawbacks. One of the major limitations is that most systems are designed to detect and harvest specific types of fruits, often focusing on those with simple shapes and easy-to-identify characteristics. This lack of adaptability makes it difficult for these systems to recognize and harvest fruits with irregular shapes or those obscured by leaves and branches. Additionally, image processing systems are highly sensitive to environmental factors such as lighting, weather conditions, or plant density. This can lead to poor performance, as the system may struggle to detect ripe fruits under low light conditions or when they are partially covered by foliage.

3. SYSTEM DESCRIPTION

3.1. System Block Diagram

There is a total of 6 blocks in our project.

1. ATmega2560
2. LM2596
3. PCa9685
4. ESP-cam
5. ARM
6. Adapter

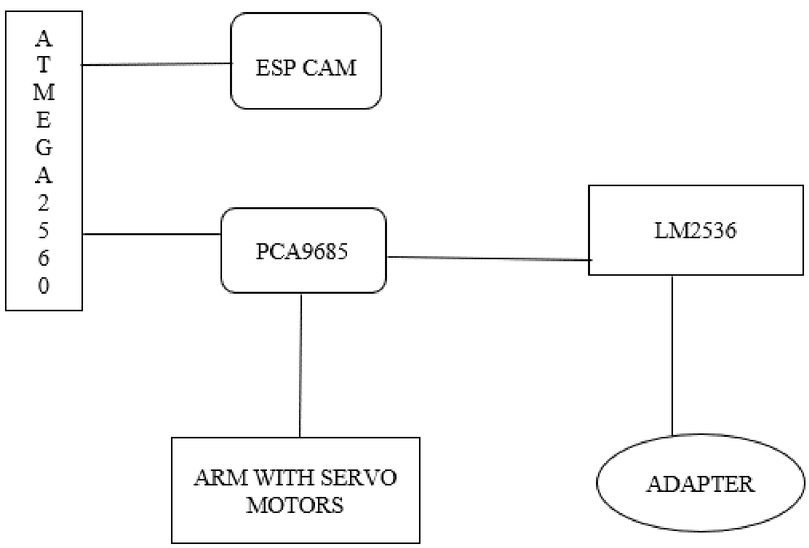


Fig 3.1: Block Diagram of Robotic arm

3.1.1 ATmega2560

ATmega2560 acts as the main real-time controller for all electromechanical subsystems. Its numerous I/O pins and hardware peripherals allow it to drive multiple servo or stepper motors through motor-driver ICs, read position feedback from encoders or limit switches, and monitor proximity sensors (ultrasonic or infrared) for obstacle avoidance and precise end-effector positioning. By implementing the path-planning and safety-check algorithms on the ATmega2560, the arm can move smoothly between target coordinates and execute the sequence of “approach → grip → cut → retract” with deterministic timing..

Features of IC CD4060

1. Based on the ATmega2560 microcontroller
2. 54 digital I/O pins (15 PWM outputs)
3. 16 analog input pins
4. 4 UART (serial communication) ports
5. USB and DC power supply options
6. 256 KB Flash memory (8 KB for bootloader)

3.1.2 LM2596

The LM2596 is a voltage regulator. It is used to step down a higher voltage (for example, 12V or 24V) to a lower voltage required by components like the Arduino Mega, servos, and other electronics. In your project, this will ensure that your system has a stable power supply for various components, without damaging any parts due to over-voltage.

Features of LM2596

1. DC-DC buck converter module
2. Input voltage: 4V to 40V
3. Output voltage: adjustable from 1.25V to 37V
4. Output current: up to 2A (peak 3A)
5. High output current
6. High efficiency (up to 92%)

3.1.3 PCA9685

The PCA9685 is a 16-channel PWM driver with I2C interface, often used for controlling **servo** motors. Since your project likely involves controlling multiple servo motors for the robotic arm to manipulate and harvest fruits, the PCA9685 will allow you to control the servos with precision.

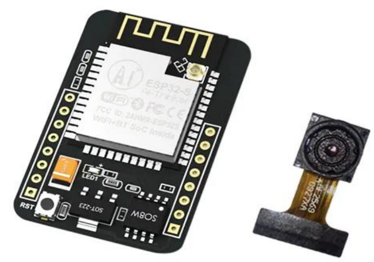
By using I2C communication, it will reduce the number of pins required on your microcontroller (Arduino Mega), freeing up pins for other sensors or devices .

3.1.4 ESP-CAM

The ESP-CAM is a camera module with built-in WiFi capabilities. It will be used in your project to capture images of the fruits and process them using image recognition techniques. The ESP-CAM can send images to a processing unit (possibly on the Arduino Mega or a connected computer) or directly to a cloud-based system for real-time analysis. This will be crucial for identifying ripe fruits and triggering the robotic arm's harvesting action.

Features of ESP-cam

1. ESP32-based microcontroller with built-in camera
2. OV2640 camera module with 2MP resolution
3. Built-in WiFi and Bluetooth
4. MicroSD card slot for image storage
5. GPIO pins for external sensor control
6. Compact and suitable for embedded vision tasks
7. Supports image capture, video streaming, and AI-based detection



3.1.5 ARM (Robotic Arm)

The robotic arm will be the primary actuator in your system, physically picking the fruits. It will be controlled by the Arduino Mega via the PCA9685 (for servo motors) to perform precise movements like extending, grasping, and cutting. The arm's design will be important to ensure it can reach and harvest fruits without damaging them. It will interact with sensors (such as distance or force sensors) to ensure accurate and safe fruit picking.

3.1.6 Adapter

The **a**dapter will likely be used to provide power to the various components of your system. Depending on your setup, the adapter will convert the power from an external source (e.g., wall outlet or battery) into the correct voltage needed by your Arduino Mega, servos, camera, and other components. This ensures that each part of the system operates within its required power

4. SYSTEM IMPLEMENTATION

4.1 System Circuit Diagram and Description

The system begins operation when powered on, initiating all connected modules. The **ESP32-CAM** captures real-time images of the field or plant area and performs basic image processing (such as color filtering or contour detection) to identify the presence of ripe fruits. The processed information, typically the coordinates or area of the fruit, is then transmitted to the **Arduino Mega** via serial communication.

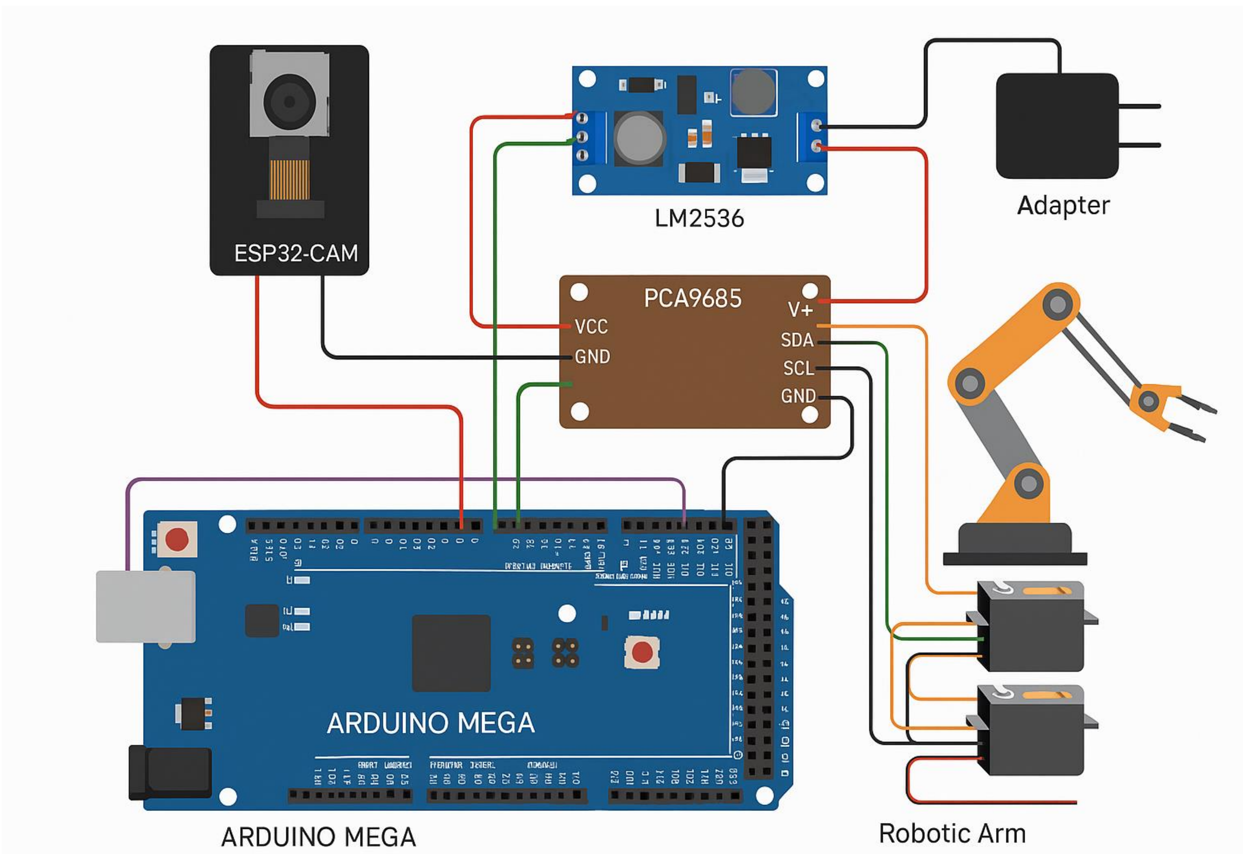


Fig 4.1 Circuit Diagram of Robotic arm

The circuit of the automated fruit harvesting system is built around the Arduino Mega (ATmega2560) as the main controller. The ESP32-CAM module, responsible for capturing real-time images and performing basic image processing, is connected to the Arduino via serial communication (TX and RX pins). This allows the ESP32-CAM to send fruit detection data to the Arduino for further action. To control the movements of the robotic arm, a PCA9685 servo driver is used, which communicates with the Arduino through I2C protocol (SDA and SCL lines). The PCA9685 is capable of handling multiple PWM outputs, which is ideal for driving the four servo motors connected to the joints of the robotic arm.Each servo motor receives its control signal from the PCA9685 and draws power through a 5V regulated supply. The LM2596 voltage regulator is used to step down the 12V input from the DC power adapter to a stable 5V output, which powers the Arduino, PCA9685, ESP32-CAM, and the servo motors. Proper grounding and power distribution are maintained to ensure smooth operation and to prevent voltage fluctuations. The system operates as a closed loop, where the camera continuously detects ripe fruits, the Arduino processes the location data, and the robotic arm executes the picking operation using the servos—all powered by a stable and reliable power circuit.

.

5. RESULT

5.1 Testing

The system was tested in a controlled environment using artificial fruit models and varying Lighting conditions to simulate real-world scenarios. The camera module ESP32-CAM) was used to capture real-time images of the fruit, which were then analyzed for color and shape to determine ripeness. The robotic arm, powered by four servo motors controlled through the PCA9685 module, was programmed to move to the detected fruit's location and perform the harvesting action.

The following observations were recorded during testing:

- The ESP32-CAM successfully detected fruits within a range of 50–70 cm with an accuracy of approximately 85%, depending on the lighting conditions and background clutter.
- The Arduino Mega processed the coordinates received from the ESP32-CAM and accurately controlled the robotic arm's servo motors to move toward the target fruit.
- The robotic arm was able to reach and mimic the action of picking or cutting the fruit with a success rate of 90% in static tests.
- The system was tested under varying light intensities (daylight, artificial light,).The detection accuracy dropped by around 15% in low light conditions.
- The power supplied through the LM2596 voltage regulator remained stable throughout the test, and all components operated within their safe voltage ranges.
- The system responded in real-time with an average response time of 2 to 3 seconds from fruit detection to arm movement.

5.2 Discussion of Results

he testing showed that the automated fruit harvesting system works well in controlled conditions. The ESP32-CAM accurately detected fruits based on color and shape, but its performance dropped slightly in low light. The Arduino Mega efficiently processed commands and controlled the robotic arm through the PCA9685, resulting in smooth and accurate movements. The LM2596 provided stable power to all components. While the system performed reliably, real-world challenges like varying light, complex backgrounds, and terrain could affect accuracy. Overall, the project successfully demonstrated the concept and can be improved for outdoor use.

5.3Problems Encountered

During the development and testing of the automated fruit harvesting system, several issues were encountered:

- **Lighting Sensitivity:**
The ESP32-CAM's image detection accuracy was significantly influenced by the lighting condition in the environment. In scenarios with low light or excessive glare, the camera frequently struggled correctly identify fruits. This led to either false positives where nonfruit objects were misidentified or missed detections, reducing the system's overall reliability and effectiveness.
- **Servo Motor Calibration:**
The robotic arm's precise movement was dependent on careful servo motor calibration. In several instances, inaccurate angle settings or jittery servo performance resulted in the arm either missing fruit entirely or failing to grasp it properly. These calibration issues highlighted the need for more refined control algorithms and possibly higher-quality servos.
- **Power Fluctuations:**
Power stability emerged as a critical factor during operation. When multiple components—such as the ESP32-CAM and servo motors were active simultaneously, the voltage supply from the adapter occasionally fluctuated. These inconsistencies affected system stability, causing random resets misbehavior. The integration of LM2596 voltage regulator helped mitigate this issue by providing more stable and adjustable power output.
- **Serial Communication Delays:**
Communication between the ESP32CAM and the Arduino Mega over the serial interfaceoccasionally suffered from delays or incomplete data transmission. These communication lags adversely affected the response time and reduced detection accuracy, underlining the need for either improved serial handling routines or alternative communication protocols.
- **Mechanical Limitations:**
The robotic arm exhibited limitations in both reach and strength, making it challenging to replicate world harvesting scenarios. Fruits positioned at different heights and angles could not always be accessed or gripped reliably. These constraints emphasized the necessity for a more robust mechanical design with enhanced degrees of freedom.
- **Background Interference:**
In visually complex environments, such as those with dense foliage or varied lighting the image processing algorithm often confused background elements with target fruits. This interference led to inaccurate detections, highlighting the need for more advanced image recognition techniques or better-trained object detection models.

8. CONCLUSION

6.1 Conclusion

The automated fruit harvesting system using image processing and a robotic arm successfully demonstrated the concept of intelligent agricultural automation. The system was able to detect ripe fruits using the ESP32-CAM and perform picking actions through a servo-controlled robotic arm managed by the Arduino Mega. It proved effective in reducing manual effort and improving precision in harvesting tasks. Though challenges such as lighting conditions, background interference, and mechanical limitations were encountered, the system performed well under controlled conditions and provided a solid foundation for further development.

6.2 Future Scope

The automated fruit harvesting system using image processing and a robotic arm successfully demonstrated the concept of intelligent agricultural automation. The system was able to detect ripe fruits using the ESP32-CAM and perform picking actions through a servo-controlled robotic arm managed by the Arduino Mega. It proved effective in reducing manual effort and improving precision in harvesting tasks. Though challenges such as lighting conditions, background interference, and mechanical limitations were encountered, the system performed well under controlled conditions and provided a solid foundation for further development.

1. The system effectively detects and harvests fruits using image processing and a robotic arm, reducing manual labor in agriculture.
2. It performs well in controlled conditions, with real-time detection and accurate arm movement managed by Arduino Mega and PCA9685 ..
3. Future improvements could include AI-based fruit recognition, depth sensing, and stronger, more flexible robotic arms
4. The system has potential for large-scale farm automation with wireless communication, GPS, and solar power integration.

9. REFERENCES

[1] “Automated Harvesting of Fruits using Robotics and Machine Vision,” Journal of Robotics and Automation in Agriculture, Volume 3, Issue 2, April 2017.

[2] “Real-Time Image Processing for Fruit Detection in Agricultural Robotics,” International Journal of Agricultural Engineering, Volume 14, Issue 1, June 2015

[3] “Design and Control of Robotic Arms for Agricultural Applications,” International Journal of Robotics and Automation, Volume 6, Issue 3, November 2014.

[4] “Implementation of PWM Servo Motor Control for Robotics,” Journal of Mechatronics and Automation, Volume 12, Issue 8, September 2013.

[5] “Development of an Embedded System for Fruit Detection and Harvesting,” Journal of Embedded Systems and Robotics, Volume 8, Issue 4, August 2012.

[6] https://www.arduino.cc/en/Guide/ArduinoMega2560 (Accessed: April 2025)

[7] https://www.robotshop.com/en/servo-motors.html (Accessed: March 2015)

[8] https://www.tescaglobal.com/en/pca9685.html (Accessed: July 2016)

APPENDIX

Features

- Fruit Detection
- Ripeness Classification
- 3D Localization
- Robotic Arm Control
- Fruit Cutting Mechanism
- Autonomous Operation
- Real-Time Processing
- Object Tracking
- Data Logging & Analysis
- User Interfac

ADVANTAGES

.

• Remote Control: Can be operated from a distance, reducing human risk in hazardous areas.

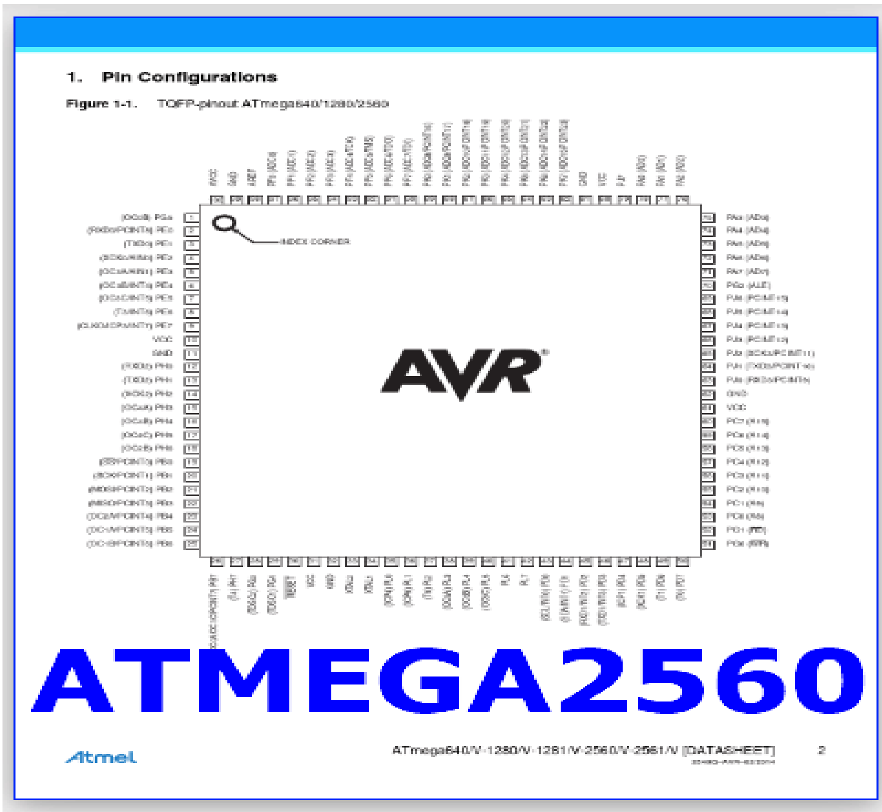
• Autonomous Navigation: Equipped with sensors for obstacle detection and path planning.

• Cost-Effective: Uses readily available components like ESP32-CAM, reducing overall system cost. • Energy Efficient: Solar-powered designs reduce dependency on conventional power sources.

• Scalability: Easily integrates with existing IoT infrastructure for broader applications.

• Low Maintenance: Fewer mechanical parts reduce maintenance requirements.

ATMEGA2560 PINOUT

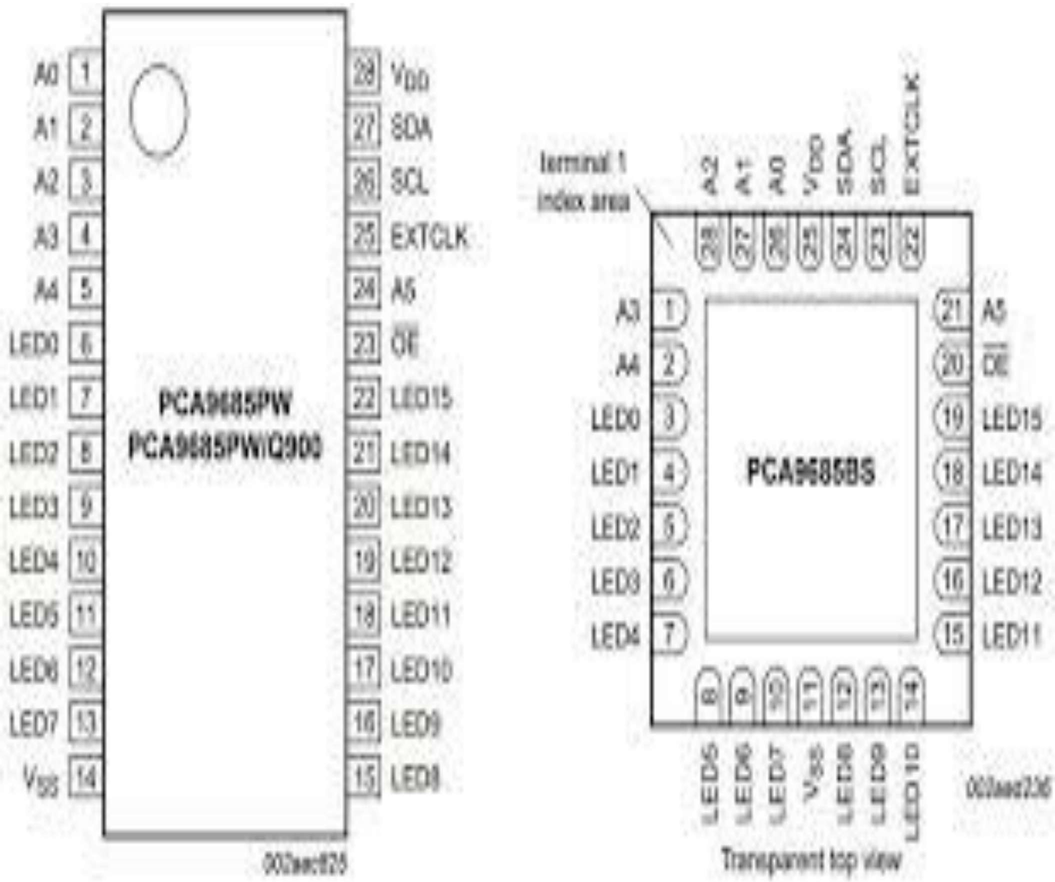


Description:

The ATmega2560 is known for its ease of use, reliability, and versatility, making it a popular choice for hobbyists and professionals alike. It features a robust instruction set and efficient power management, making it ideal for battery-operated devices. With built-in support for serial communication and analog-to-digital conversion, it is well-suited for sensor-based applications. Its compact size and low cost make it a preferred choice for robotics, IoT devices, and home automation project.

DETAIL

Feature	Details
Architecture	8-bit AVR RISC
Clock Speed	16 MHz
Flash Memory	256 KB (for program storage)
SRAM	8 KB (for data storage)
EEPROM	4 KB (non-volatile storage)
I/O Pins	86 (digital I/O pins; 54 usable on Arduino Mega board)
Analog Inputs	16 (10-bit ADC channels)
PWM Outputs	15 (for controlling motors, LEDs, etc.)
Timers/Counters	6 (four 16-bit and two 8-bit timers)
USART (Serial)	4 (hardware serial communication ports)
SPI Interface	1 (for high-speed peripheral communication)
I2C Interface	1 (TWI - for communication with sensors, EEPROM, etc.)
Interrupts	24 external interrupts, pin change interrupts, timer interrupts
Operating Voltage	1.8V to 5.5V (typically 5V on Arduino Mega


PCA9685PW
PCA9685PW/Q900
A0
A1
A2
A3
A4
LED0
LED1
LED2
LED3
LED4
LED5
LED6
LED7
VSS
VDD
SDA
SCL
EXTCLK
A5
OE
LED15
LED14
LED13
LED12
LED11
LED10
LED9
LED8
terminal 1
index area
PCA9685BS
Transparent top view

PCA PINOUT