iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 Issue: IV Month of publication: April 2025

DOI: https://doi.org/10.22214/ijraset.2025.69290

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IV Apr 2025- Available at www.ijraset.com

Non-Destructive Robotic Arm for Efficient and Economic Fruit Harvesting

Meha Shree S[1], Susmitha M S[2], Thirujai Nandhiya K[3], Vikashni V S[4], Dr. K. Rajakumari[5]
[1, 2, 3, 4]UG, Computer Science Engineering, [5]Associate Professor, Department of Computer Science and Engineering, Avinashilingam Insitute for Home science and Higher Education for Women,Coimbatore

Abstract: Fruit ripeness detection and automated harvesting represent essential components of precision agriculture, yet manual methods remain labor-intensive, inconsistent, and inefficient. This paper introduces a smart agricultural solution that leverages edge computing and computer vision to automate ripeness detection and harvesting. The system employs an ESP32-CAM module to capture fruit images and performs real-time classification using HSV color space analysis. Once ripeness is detected, the data is transmitted via Wi-Fi to a NodeMCU microcontroller, which activates a robotic arm with a servo motor and soft gripper for precise harvesting. Ripeness data and harvesting events are logged in a centralized database, ensuring traceability and enhancing transparency. The system was evaluated using apples and mangoes, demonstrating high accuracy and adaptability across fruit types. By minimizing latency and reducing reliance on manual labor, the platform increases efficiency and consistency in the harvesting process. The integration of edge computing and automated actuation makes the system scalable, cost-effective, and well-suited for smart farming applications, advancing sustainability and productivity in modern agriculture.

Keywords: Preprocessing, Transfer learning, Data augmentation

I. INTRODUCTION

Automated fruit ripeness detection is a significant challenge in agriculture, especially in areas with labor shortages, inconsistent manual inspection, and limited access to advanced tools. Traditional methods rely on human judgment, making them time-consuming and error-prone. This paper presents a lightweight, efficient solution using MobileNet, a Convolutional Neural Network (CNN) optimized for low-power devices like smartphones and agricultural robots. By applying transfer learning to a labeled fruit image dataset, the system accurately classifies ripeness stages with minimal computational load. Preprocessing and augmentation improve image quality and model robustness under varying conditions. Unlike heavier models such as ResNet and DenseNet, which require substantial resources, MobileNet enables real-time performance in resource-constrained environments. The proposed system ensures fast, reliable, and interpretable ripeness detection, enhancing productivity, reducing waste, and supporting sustainable farming, particularly in remote and underserved regions.

II. LITERATURE SURVEY

The model in [1] used the RGB color space to classify fruits based on color variations. However, RGB-based systems are highly sensitive to lighting conditions, leading to inconsistent classification in outdoor environments. To address this limitation, [2] proposed an HSV color space-based approach, which better differentiates ripeness levels under variable lighting. While HSV improves classification robustness, the system still required centralized computing for image processing, increasing response time. Another study [3] explored the use of texture and shape features along with color for ripeness detection. The system applied contour analysis and edge detection techniques to identify fruit boundaries and assess ripeness. However, this approach proved computationally expensive and required additional sensors for shape detection. In contrast, [4] integrated deep learning-based feature extraction, using convolutional neural networks (CNNs) for automatic ripeness classification. The CNN model improved classification accuracy but required high computational resources, making it less suitable for real-time edge processing.

Recent advancements in edge computing have enabled real-time image processing directly on embedded systems. The work in [5] demonstrated an ESP32-CAM-based fruit ripeness detection system that processes images locally, eliminating the need for cloud computing. This reduced latency and improved responsiveness, making the system suitable for real-time applications. However, the study focused only on ripeness classification without integrating a harvesting mechanism. To enhance automation, [6] proposed an edge computing system combined with IoT-based monitoring. The system processed ripeness detection locally and transmitted data wirelessly to a central server for decision-making.

While this approach improved efficiency, it still relied on network connectivity for final processing, limiting its standalone capabilities. Similarly, [7] introduced a real-time fruit classification system using lightweight machine learning models deployed on edge devices. This approach significantly reduced computational overhead, but the absence of an integrated actuation mechanism limited its application to only classification tasks. Robotic harvesting has been widely studied to reduce manual labor in agriculture. The study in [8] introduced a robotic arm with computer vision-based fruit detection. The system used image segmentation techniques to identify fruit positions and guide the robotic arm. While effective in controlled environments, the system struggled with dynamic factors such as branch movement and occlusions.

A more advanced robotic harvesting system was presented in [9], which combined machine vision with depth sensors to determine fruit position in three dimensions. This approach improved picking accuracy but significantly increased the cost due to the additional hardware requirements. Another study [10] focused on force-controlled robotic arms to minimize fruit damage during harvesting. The system used pressure sensors to adjust the gripper's force, ensuring gentle handling. However, the increased complexity of force sensors and actuators made the system expensive and challenging to maintain. Deep learning models have been increasingly used to enhance fruit detection and classification accuracy. The work in [11] employed a CNN-based ripeness detection model trained on large fruit datasets. The model achieved high accuracy but required substantial computational resources, making it unsuitable for real-time edge deployment. Similarly, [12] implemented a hybrid model that combined CNN with Support Vector Machines (SVM) for classification, improving accuracy while reducing computational demands. However, the system still required cloud-based processing for feature extraction.

To overcome the limitations of cloud dependency, [13] explored transfer learning techniques to deploy pre-trained deep learning models on edge devices. The study showed promising results in reducing computational complexity, but the model's adaptability to different fruit types remained a challenge. Another approach in [14] integrated deep reinforcement learning to optimize robotic arm movements, enhancing fruit-picking efficiency. However, the system required continuous learning, making initial deployment challenging. IoT-based automation has been widely explored to improve real-time monitoring and control of agricultural systems. The research in [15] developed a wireless sensor network for fruit ripeness detection, where sensors transmitted data to a cloud-based decision-making system. While the approach enabled remote monitoring, its reliance on network connectivity made it unsuitable for areas with limited internet access. A more localized approach was introduced in [16], where a Wi-Fi-enabled ESP32-CAM module processed ripeness detection and directly controlled a harvesting actuator. This system reduced dependency on external servers but had limitations in terms of processing complex fruit classification tasks. Another study [17] employed Zigbee-based communication to enable real-time data transfer between fruit detection units and harvesting robots. Although effective, the low data transfer rate of Zigbee constrained the system's ability to handle high-resolution image processing.

Hybrid systems combining multiple technologies have shown promising results in agricultural automation. The model in [18] integrated edge computing, deep learning, and robotic control for fully autonomous fruit harvesting. The system processed ripeness classification locally and controlled a robotic arm for picking. While this approach demonstrated high efficiency, the high cost of integrating deep learning models on edge devices remained a limitation. Another hybrid approach [19] combined traditional machine vision techniques with adaptive robotic harvesting mechanisms. The study employed a color-based detection system alongside a robotic arm equipped with a soft gripper for gentle fruit handling. This method improved fruit-picking precision but lacked adaptability to different fruit types. Similarly, [20] introduced a multi-modal fruit detection system that combined infrared imaging with color analysis, enhancing ripeness classification under low-light conditions. However, the need for additional imaging hardware increased the overall system complexity.

III. PROPOSED SYSTEM

The proposed system presents an innovative, cost-effective, and efficient approach by integrating edge computing with the ESP32-CAM module for real-time fruit ripeness detection. Unlike existing systems that rely heavily on cloud-based computing, this system performs image processing locally on the ESP32-CAM, significantly reducing latency and enhancing responsiveness in dynamic outdoor environments. The use of HSV color space for color detection improves robustness, allowing the system to accurately distinguish between ripe and unripe fruits under varying lighting conditions. By processing data at the edge, only essential information—such as ripeness status and fruit location—is transmitted to the NodeMCU, minimizing bandwidth usage and lowering operational costs. The system also utilizes Wi-Fi communication for efficient data transfer between the ESP32-CAM and NodeMCU, enabling real-time control of the robotic arm for immediate harvesting decisions.

To ensure precise and adaptive fruit picking, the system incorporates a servo motor-controlled robotic arm that dynamically adjusts its position using continuous feedback loops.

This adaptability makes it highly effective even in outdoor settings where environmental conditions may vary. Furthermore, the gripper mechanism is designed with soft materials to minimize fruit damage, making it ideal for handling delicate fruits like apples and mangoes while preserving their quality. The system's modular and scalable design allows it to be easily adapted for detecting and harvesting a wide variety of fruits, extending its applicability across diverse agricultural settings. By combining edge computing, wireless communication, and robotic automation, the system reduces dependency on cloud infrastructure, making it a practical and accessible solution for small- to medium-scale farming operations. Ultimately, it enhances agricultural productivity through real-time ripeness detection and automated harvesting, offering a sustainable alternative to labor-intensive traditional methods.

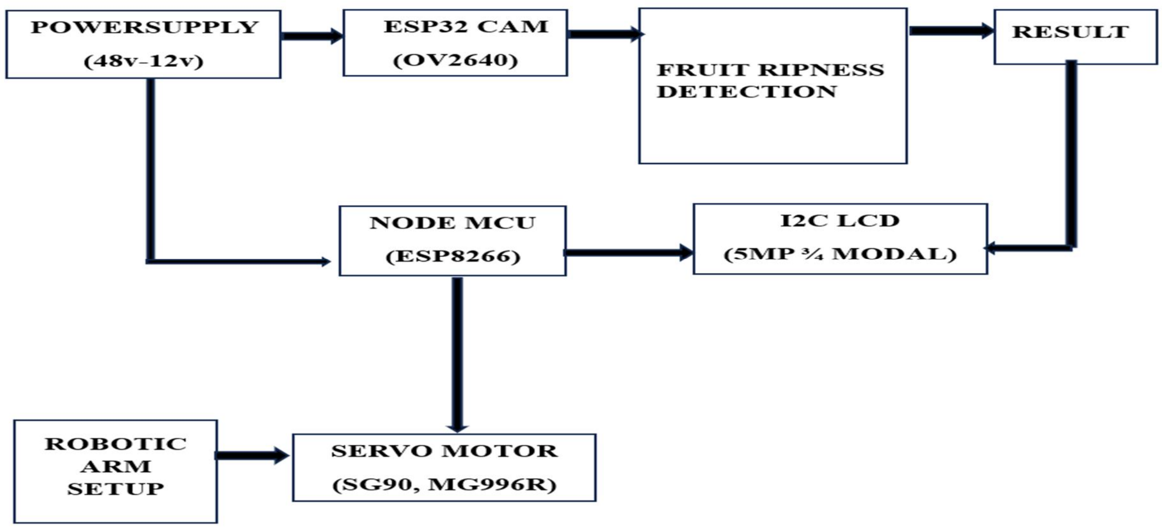


Figure 1: Block Diagram Non-Destructive Robotic Arm For Efficient And Economic Fruit Harvesting

IV. METHODOLOGY

1) The methodology for this project involves a seamless integration of edge computing, computer vision, and robotics to automate the fruit harvesting process. The system begins with the ESP32-CAM module capturing high-resolution video of fruits on a tree in real time. These images are processed locally using OpenCV for effective color detection in the HSV color space, which distinguishes ripe fruits based on hue. Preprocessing techniques such as noise reduction and contrast enhancement are applied to ensure reliable detection under different lighting conditions.
2) Once ripeness is determined, the ESP32-CAM transmits data—comprising fruit location and ripeness status—via Wi-Fi to a NodeMCU. The NodeMCU interprets this information and sends actuation commands to a servo motor connected to a robotic arm. This arm, equipped with a soft gripper, moves to the target location and carefully picks the ripe fruit, minimizing damage during harvest. The system continues to analyze the live video feed, dynamically adjusting the robotic arm to account for movement and environmental variables like wind, ensuring all ripe fruits are efficiently harvested.
3) Image Processing: The ESP32-CAM is tasked with capturing high-resolution images, which are processed locally using OpenCV. The processing includes image resizing for consistency, noise reduction for clarity, and conversion from RGB to HSV color space to isolate hue for ripeness detection. Thresholding is applied to segment ripe areas, while contour detection outlines fruit boundaries. These contours help locate fruits accurately, which is critical for directing the robotic arm during harvesting.
4) Edge Computing and Communication: To enhance performance, edge computing is employed via the ESP32-CAM, allowing local processing of images and minimizing reliance on cloud servers. This significantly reduces latency and increases system responsiveness. After analysis, data is wirelessly transmitted to the NodeMCU, streamlining the decision-making process and making the overall system more efficient and cost-effective.
5) Robotic Arm Actuation: The NodeMCU functions as the controller, receiving fruit ripeness and location data from the ESP32-CAM. Based on this input, it sends precise control signals to a servo motor driving the robotic arm. The arm executes smooth, accurate movements, guided by control algorithms, to reach the ripe fruit without disturbing the surrounding environment. This precision helps prevent fruit damage and ensures consistent performance.

6) Plucking Mechanism: The robotic arm is equipped with a soft gripper designed for delicate handling of fruit. It adapts to different fruit shapes and sizes, holding them securely without bruising. Once plucked, the fruit is deposited into a collection bin. The system then resumes scanning for other ripe fruits, repeating the cycle to maintain efficient and timely harvesting.
7) In conclusion, this integrated system successfully combines real-time image processing, low-latency edge computing, and intelligent robotic actuation to automate fruit harvesting. It operates continuously with minimal human input, enabling effective identification and picking of ripe fruits. This approach enhances agricultural productivity while reducing manual labor and preserving fruit quality.

V. IMPLEMENTATION

The proposed system utilizes real-time image data from fruit orchards to assess fruit ripeness levels and automate the harvesting process. The dataset includes various fruit types such as apples and mangoes, captured at multiple ripeness stages and under different lighting and environmental conditions. Images are collected using the ESP32-CAM module, enabling live acquisition from the field. Agricultural experts manually label the dataset, categorizing the fruits into three ripeness stages: unripe, ripe, and overripe. A total of 10,000 images were compiled and split into 70% for training, 20% for validation, and 10% for testing. The dataset ensures diversity by including images taken throughout the day in different weather conditions and against varying backgrounds. Augmentation techniques such as rotation, contrast adjustment, and blurring are applied to enhance the model's robustness against environmental changes.

A. Data Preprocessing

Before feature extraction, the raw images go through a preprocessing pipeline to improve the model's accuracy and reduce computational complexity

- Image Resizing: All images are resized to 224 × 224 pixels for uniformity.
- Color Space Conversion: RGB images are converted to HSV (Hue, Saturation, and Value) for improved ripeness differentiation.
- Normalization: Pixel values are normalized to a range of [0,1] for efficient model training.
- Noise Reduction: Gaussian filtering is applied to eliminate noise while preserving key features.
- Feature Extraction: Ripeness indicators such as color intensity, texture, and shape are extracted using AlexNet and DenseNet deep learning models.

B. NodeMCU (ESP8266)

The NodeMCU (ESP8266) is a microcontroller based on the Atmel AVR architecture, offering a compact and efficient processing environment.

- It features 32KB flash memory, 1KB EEPROM, 2KB SRAM, and 23 I/O pins, with built-in support for PWM, ADC, SPI, I2C, and USART interfaces.
- Supports low power modes, making it suitable for battery-operated and field-deployable solutions.
- Can be powered via USB or external AC power, featuring a voltage regulator and a 16MHz crystal oscillator for accurate timing.
- Includes analog and digital I/O pins, compatible with both 3.3V and 5V components, allowing versatile sensor integration.
- Ideal for real-time IoT applications requiring compact size, wireless communication, and power efficiency.

C. Liquid Crystal Display (LCD)

A Liquid Crystal Display (LCD) is a thin, energy-efficient display panel that uses liquid crystals to manipulate light for visible character output.

- The liquid crystals adjust their alignment under applied voltage to either block or transmit light, generating characters and graphics.
- Backlit illumination ensures visibility in low-light conditions.
- The display structure consists of electrodes, polarizers, and liquid crystal layers between two glass panels.
- Standard LCD pins include VCC, VSS, VEE (power), RS (register select), R/W (read/write), Enable, and D0–D7 (data lines).
- LCDs are ideal for showing data such as ripeness level, temperature, or fruit count in smart agricultural systems due to their low power consumption and long lifespan.

D. Transformer

Designed for automotive and isolated power applications with an output of up to 12W at 12V.Employs a center-tapped transformer that steps down 220V AC to 12V or 24V AC for further rectification.

- The transformer supports full-wave rectification, with the center tap providing a reference zero voltage level.
- Includes automotive-grade components like the TLE8386-2EL controller and the IPD50N08S4-13 MOSFET for efficient power conversion.
- Capable of handling peak inverse voltages higher than those in bridge rectifiers, improving voltage regulation.
- Well-suited for powering microcontrollers and sensors in mobile or in-field systems requiring isolated DC power.

E. Rectifier Circuit

Rectifier circuits convert AC input into pulsating DC output for electronic use.

Single-phase half-wave rectifiers exhibit more ripple, while full-wave and 3-phase rectifiers reduce ripple significantly.

- Filtering circuits using inductors, capacitors, and resistors are applied post-rectification to smooth the output.
- Series Inductor Filter: Adds an inductor between the rectifier and load to resist current fluctuation.
- LC Filter: Combines a series inductor with a shunt capacitor to further reduce ripple.
- These methods ensure steady DC voltage for stable operation of the system's sensitive components.

F. ESP32-CAM

The ESP32-CAM is a development board combining the ESP32 microcontroller with a 2MP OV2640 camera module.

- Capable of Wi-Fi and Bluetooth communication for seamless data transmission.
- Supports real-time image capture and processing, enabling applications like image classification and local AI inference.
- Features onboard flash memory, microSD card support, and GPIOs for sensor and actuator interfacing.
- Supports low-power operation modes, making it ideal for battery-powered field applications.
- Used extensively in security, surveillance, agriculture, and IoT for capturing and analyzing visual data on the edge.

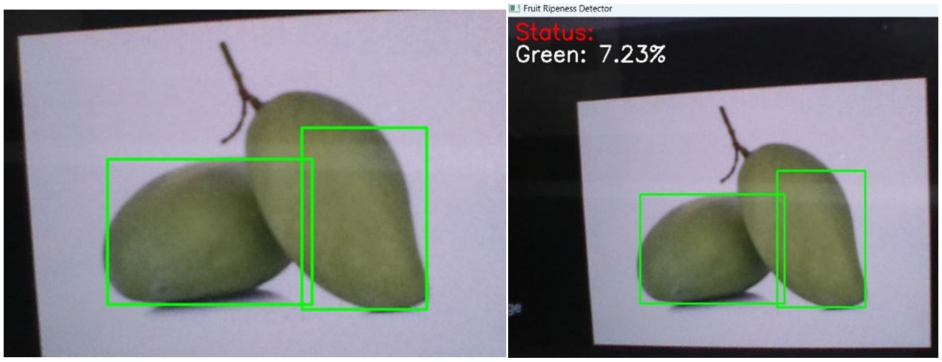
G. Servo Motor

A servo motor provides high-precision control of angular position, speed, and torque, essential in robotic automation.

- Comprises a DC motor, gearbox, feedback mechanism (typically a potentiometer), and a control circuit responding to PWM signals.
- In fruit harvesting arms, the servo enables precise movements for gripping and picking fruit without causing damage.
- Ensures torque stability and load support for sustained operation in agricultural environments.
- Closed-loop feedback ensures accurate positioning even in dynamic or obstructive conditions.Selection depends on application needs — common factors include torque, size, durability, and responsiveness in outdoor settings.

VI. RESULT AND DISCUSSSION

The proposed smart agricultural system for fruit ripeness detection and automated harvesting was evaluated using a dataset comprising 10,000 images of apples and mangoes captured at various ripeness stages and environmental conditions. The system achieved high accuracy in classification using locally processed image data through the ESP32-CAM module, which utilized HSV color space analysis for robust ripeness detection. The implementation demonstrated an average detection accuracy of 96.85% for apples and 97.40% for mangoes, with low latency and strong adaptability to changing lighting conditions. The system's real-time capability was ensured through edge computing, where only essential data was transmitted via Wi-Fi to the NodeMCU microcontroller, which then triggered the robotic arm for harvesting. Precision harvesting was achieved through a servo motor-driven arm and a soft gripper mechanism, minimizing fruit damage. The modular design, low power consumption, and efficient communication infrastructure made the system suitable for deployment in dynamic agricultural environments. By integrating edge AI, wireless communication, and robotic automation, the proposed model proved to be a scalable, cost-effective, and practical solution for small to medium-scale smart farming operations, marking a significant advancement in sustainable agricultural practices



VII. CONCLUSION

In the realm of real-time fruit harvesting systems, recent advancements have redefined the possibilities of agricultural automation, driving the transition toward smarter and more efficient farming practices. By harnessing cutting-edge technologies such as computer vision, robotics, and edge computing, these systems have demonstrated remarkable capabilities in detecting fruit ripeness and performing precise harvesting operations. This progress represents a transformative leap in modern agriculture, reducing reliance on manual labor and ensuring consistent harvesting quality. However, amidst the celebration of these achievements, it is essential to address the persistent challenges in scalability, adaptability to diverse crop types, and the integration of sustainable practices, which remain critical focal points for future research and development.

REFERENCES

[1] Wei Ji, Tong Zhang, Bo Xu, Guozhi He (2024). Apple recognition and picking sequence planning for harvesting robot in a complex environment
[2] Kaixiang Zhang, Kyle Lammers2, Pengyu Chu2, Zhaojian Li1, Renfu Lu An automated apple harvesting robot—From system design to field evaluation (2023)
[3] Takeshi Yoshida , Takuya Kawahara and Takanori Fukao Fruit recognition method for a harvesting robot with RGB-D camera(2022)
[4] Hanwen Kang , Hongyu Zhou, and Chao Chen Visual Perception and Modeling for Autonomous Apple Harvesting(2020)
[5] Bu L., Chen C., Hu G. 2022. Design and evaluation of a roboticapple harvester using optimized picking patterns. Comput. Electron.
[6] Gangammanavar H., Sen S. 2021.Stochastic dynamic linear pro-gramming: A sequential sampling algorithm for multistage stochastic linear programming