VISION ASSISTED ROBOTIC ARM USING ARDUINO

Submitted in partial fulfilment of the requirements of the degree of

Bachelor of Engineering

by

Khan Mohammad Tahir (Roll No. 18)

Under the guidance of

Prof. Monika Pathare

University of Mumbai

Department of Computer Engineering,

Theem College Of Engineering

Village Betegaon, Boisar Chilhar Road, Boisar (E), Palghar

2023-2024

VISION ASSISTED ROBOTIC ARM USING ARDUINO

Submitted in partial fulfilment of the requirements of the degree of

Bachelor of Engineering

by

Khan Mohammad Tahir (Roll No. 18)

Under the guidance of

Prof. Monika Pathare

Department of Computer Engineering,

Theem College Of Engineering

Village Betegaon, Boisar Chilhar Road, Boisar (E), Palghar

University Of Mumbai

2023-2024

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included; i have adequately cited and referenced the original sources. I also declare that i have adhered to all principals of academics honestly and integrity have not misrepresented or fabricated or falsified any idea/data/fact/sources in my submission. I understand that any violation of the above will be cause for disciplinary action by the institute and can also evoke penal action from the source which has thus not been properly cited or from whom proper permission has not been taken when needed.

Khan Mohammad Tahir 18

Date:

Project Report Approval for Bachelor Of Engineering

The project report entitled **Vision Assisted Robotic Arm Using Arduino** by Khan Mohammad Tahir is approved for the degree of Bachelor of Engineering in Computer Engineering.

Date:

Place:

Examiners:

Name & Sign of External Examiner

Name & Sign of Internal Examiner

CERTIFICATE

This is to certify that the project entitled **"Vision Assisted Robotic Arm Using Arduino"** is a bonafide work of **"Khan Mohammad Tahir (18)"** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **"Bachelor of Engineering" in "Computer Engineering"** has been carried out under my supervision at the department of Computer Engineering of Theem College of Engineering, Boisar. The work is comprehensive, complete and fit for evalautaion.

Prof. Monika Pathare
Project Guide

Prof. Shahegul Afroz
Project Coordinator

Prof. Mubashir Khan
HOD

Dr.Riyazoddin S
Principal

Acknowledgement

First and foremost, i thank God Almighty for blessing me immensely and empowering me at times of difficulty like a beacon of light. Without His divine intervention i wouldn't have accomplished this project without any hindrance.

I also grateful to the Management of Theem College of Engineering for their kind support. Moreover, i thank my beloved Principal **Dr.Riyazoddin S**, our Director, **Dr.N.K. Rana** for their constant encouragement and valuable advice throughout the course.

I profoundly indebted to **Prof. Mubashir Khan**, Head of the Department of Computer Engineering and **Prof. Shahegul Afroz**, Project Coordinator for helping me technically and giving valuable advice and suggestions from time to time. They are always our source of inspiration.

Also, i would like to take this opportunity to express my profound thanks to my guide **Prof. Monika Pathare**, Assistant Professor, Computer Engineering for her valuable advice and whole hearted cooperation without which this project would not have seen the light of day.

I express my sincere gratitude to all Teaching/Non-Teaching staff members of Computer Engineering department for their co-operation and support during this project.

Khan Mohammad Tahir(18)

ABSTRACT

This report presents the design, development, and implementation of a vision-assisted robotic arm system for the purpose of object sorting based on color. The project combines the realms of robotics, computer vision, and automation to achieve precise and efficient sorting of objects by their visual characteristics. The primary goal of this work was to create a robotic system capable of recognizing and differentiating objects by color and subsequently placing them into separate bins. The project's methodology involved the assembly and integration of a robotic arm with a computer vision system. The computer vision component utilized OpenCV for color detection and object recognition. The robotic arm, controlled by an Arduino microcontroller, was equipped with an end-effector capable of grasping and moving objects based on the color information provided by the vision system. Through a series of experiments and testing, the system demonstrated a high degree of accuracy in identifying and sorting objects by color. The results reveal the effectiveness of the color recognition algorithms and the reliability of the robotic arm's movements. Challenges encountered during the project, such as variations in lighting conditions and object appearances, were addressed and mitigated. This work contributes to the field of robotics and automation by providing a practical application of computer vision in the context of object sorting. The report also discusses the potential for future improvements and extensions of the system, as well as its broader implications in industries requiring automated sorting and handling of objects based on visual cues. The successful completion of this vision-assisted robotic arm project underscores the potential of integrating robotics and computer vision to enhance automation and efficiency in various applications.

List of Figures

Contents

Chapter 1

INTRODUCTION

Our vision-assisted robotic arm system leverages the capabilities of the Arduino Uno R3 microcontroller to automate object sorting based on color recognition. Our proposed system is a testament to the fusion of robotics and computer vision, driven by the Arduino Uno R3, to automate and enhance object sorting by color. By utilizing this technology stack, we aim to deliver a cost-effective and versatile solution that efficiently addresses the challenges posed by manual sorting processes across various industries. The proposed solution involves the design,

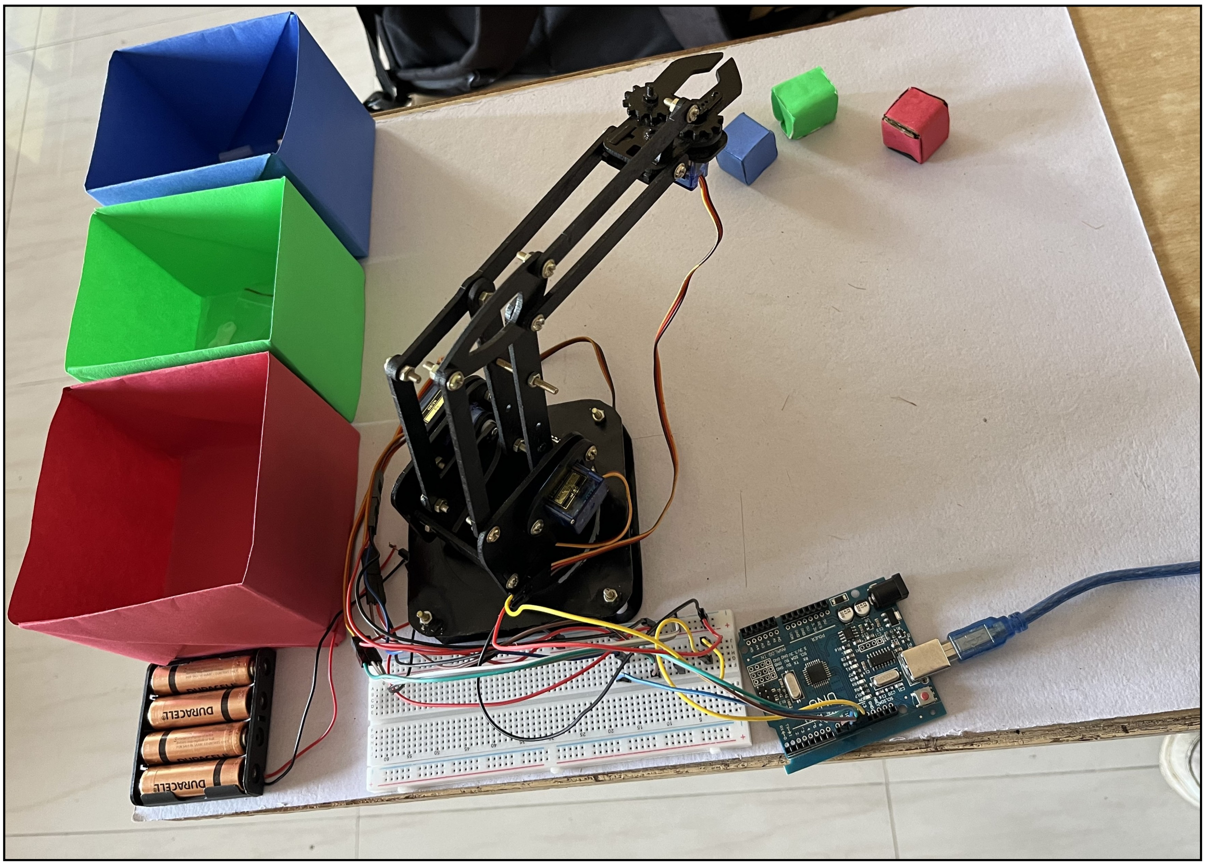

Figure 1.0.0.1: Description of the image

implementation, and optimization of an automated system using a vision-assisted robotic arm controlled by Python. By integrating computer vision capabilities, the system seeks to address the outlined challenges

In today's fast-paced industrial landscape, the demand for efficient automation solutions is ever-increasing. Traditional manual sorting processes are not only labor-intensive but also prone to errors, leading to inefficiencies and increased production costs. Recognizing the need for innovation in this domain, our vision-assisted robotic arm system integrates cutting-edge technologies to revolutionize object sorting.

By harnessing the power of computer vision and robotics, our system offers unparalleled accuracy and speed in identifying and sorting objects based on their color. The Arduino Uno R3 microcontroller serves as the backbone of our solution, providing the necessary computational power and control mechanisms to orchestrate seamless operation.

Moreover, the versatility of our system enables its application across a myriad of industries, ranging from manufacturing and logistics to recycling and agriculture. Whether it's sorting components on a production line or segregating recyclable materials, our solution adapts effortlessly to diverse operational requirements, streamlining processes and maximizing productivity.

With an unwavering commitment to innovation and excellence, our vision-assisted robotic arm system represents a paradigm shift in automation technology. By leveraging the synergies between robotics and computer vision, we aim to empower industries worldwide with a cost-effective and scalable solution that redefines the standards of object sorting efficiency.

The Arduino Uno R3 microcontroller serves as the backbone of our solution, providing the necessary computational power and control mechanisms to orchestrate seamless operation. The Arduino platform's flexibility and ease of integration allow for the rapid prototyping and development of the system, making it accessible and adaptable to various industrial scenarios.

In addition to enhancing efficiency and accuracy, our vision-assisted robotic arm system prioritizes safety and reliability. Equipped with advanced sensors and real-time monitoring, the system can detect and respond to potential obstacles or errors swiftly, minimizing risks and downtime.

With an unwavering commitment to innovation and excellence, our vision-assisted robotic arm system represents a paradigm shift in automation technology. By leveraging the synergies between robotics and computer vision, we aim to empower industries worldwide with a cost-effective and scalable solution that redefines the standards of object sorting efficiency and productivity.

Chapter 2

LITERATURE SURVEY

2.1 Literature Survey

2.1.1 "Learning from Demonstration for Robotic Arm Control with a MATLAB GUI"

Begin by discussing the concept of "Learning from Demonstration" (LfD) and how it applies to robotic arm control. Explore existing literature on LfD techniques, algorithms, and their applications in robotics. Investigate the role of Graphical User Interfaces (GUIs) in simplifying the training process for robotic arms. Examine the use of MATLAB as a platform for developing user-friendly GUIs and the advantages it offers in this context. Review studies or projects that have successfully implemented LfD in robotic arm control and assess their usability and effectiveness. Discuss the advantages and limitations of this approach in terms of ease of use, time efficiency, and scalability.

2.1.2 "Object Detection and Recognition Algorithms on a Robotic Arm Platform Using Raspberry Pi"

Explore the field of computer vision and its significance in object recognition for robotics. Review the characteristics and capabilities of various object recognition algorithms, including SIFT, SURF, FAST, and ORB. Investigate the use of the Raspberry Pi platform for implementing computer vision and object recognition in real-time applications. Examine projects or research papers that have successfully integrated object recognition with robotic arm control using Raspberry Pi. Discuss the limitations and challenges associated with using Raspberry Pi for real-time computer vision and robotics applications. Explore the adaptability of these algorithms to different scenarios and the extent to which they can be applied to diverse objects and environments.

2.1.3 "Comparative Analysis of Image Processing Implementations for Robotic Arm Control"

Begin by introducing the concept of comparative analysis in the context of image processing for robotic arm control.Explore the various image processing implementations and algorithms commonly used in robotics. Investigate different platforms, operating systems, and programming languages used for implementing image processing algorithms. Discuss the criteria and metrics used for comparing the performance of different implementations. Review studies or projects that have conducted comparative analyses of image processing implementations and present their findings. Consider the practical implications and real-world applications of the results obtained from such comparative analyses.

2.1.4 "Design and Fabrication of an Affordable 4-DOF SCARA Robotic Manipulator for Pick and Place Tasks"

Begin by discussing the significance of designing affordable robotic manipulators for educational and research purposes. Explore the concept of a 4-DOF SCARA robotic manipulator and its suitability for pick-and-place tasks. Investigate the design and fabrication process of such a robotic manipulator, including materials and components used. Examine case studies or projects that have successfully developed and utilized 4-DOF SCARA robotic manipulators. Discuss the advantages and limitations of this approach, particularly in terms of cost-efficiency and customizability. Explore potential future directions and improvements for such robotic manipulators.

2.1.5 "Object Color Recognition and Sorting Robot Using OpenCV and Machine Vision"

Begin by discussing the importance of automation and object recognition in industrial applications. Explore the field of machine vision and its role in object recognition based on color characteristics. Investigate the use of OpenCV and machine vision algorithms for color recognition and sorting. Review case studies or projects that have successfully integrated color recognition and robotic manipulation. Discuss the practical advantages of such systems in terms of automation, efficiency, and cost-effectiveness. Examine the limitations and challenges associated with lighting conditions, object angles, and scalability in real-world applications.

2.1.6 "Integration of Robotic Arm with Vision System"

In modern industrial applications, automation has become indispensable, driving efficiency and productivity. Object recognition stands as a cornerstone in this paradigm, facilitating tasks ranging from sorting to quality control. Machine vision, a pivotal aspect of automation, enables robots to discern objects accurately. Color recognition, a subset of machine vision, plays a vital role in distinguishing between objects based on their hues. Leveraging tools like OpenCV and machine vision algorithms, systems can analyze color characteristics, facilitating precise sorting and manipulation. Successful integration of color recognition with robotic manipulation has been demonstrated in various case studies and projects. For instance, in manufacturing, robots equipped with such systems streamline assembly processes by sorting products based on color. The practical advantages are abundant, with automation reducing labor costs and enhancing efficiency by eliminating manual intervention. However, challenges persist, particularly regarding lighting conditions, object angles, and scalability. Despite these hurdles, the integration of color recognition with robotic manipulation promises substantial benefits in industrial automation, paving the way for increased efficiency and cost-effectiveness.

2.1.7 "Design and simulation robotic arm with computer vision for inspection process"

In this study, a robotic arm integrated with computer vision for inspection processes is designed and simulated. Leveraging kinematics and dynamics modeling alongside MATLAB simulations, the system's behavior and performance are thoroughly analyzed. Control systems with microcontrollers are employed to ensure precise control over the robotic arm's movements during inspections. The advantages of this integration include design flexibility and material adaptability, ultimately leading to enhanced automation and efficiency in inspection tasks. However, limitations such as dependency on lighting conditions and constraints in field of view and sample size are acknowledged, necessitating further research to address these challenges for broader applicability.

Chapter 3

LIMITATIONS OF EXISTING SYSTEM

3.1 Limitation in Exiting System

3.1.1 Scalability Limitations:

- The current system struggles to accommodate increasing demands or expand its capabilities efficiently, potentially hindering future growth and scalability initiatives. Limited scalability may result in system overloads, decreased performance, and difficulties in integrating new features or accommodating higher workloads. Many systems struggle to scale efficiently as the volume of data or users increases. This can lead to performance issues and reduced system responsiveness.

3.1.2 High Maintenance Costs:

- The existing system incurs significant expenses associated with ongoing maintenance, repairs, and upgrades. High maintenance costs can strain financial resources and impede investment in other critical areas of development. Moreover, frequent maintenance requirements may disrupt operations and lead to downtime, impacting productivity. Complex or poorly designed systems can have high maintenance costs, including ongoing support, upgrades, and fixes.

3.1.3 Legacy Technology:

- The reliance on outdated or legacy technology in the current system poses several challenges, including compatibility issues with modern software and hardware, limited support from vendors, and vulnerabilities to security threats. Legacy technology may also lack essential features and optimizations found in newer solutions, hindering the system's performance and functionality. Systems built on outdated or legacy technology may lack support, face difficulties in maintenance, and be less adaptable to new requirements.

3.1.4 Lack Of Documentation:

- Inadequate or outdated documentation within the system makes it difficult for users and administrators to understand its architecture, functionalities, and configurations. The absence of comprehensive documentation can impede troubleshooting efforts, prolong downtime during maintenance, and hinder the onboarding process for new users. Clear and up-to-date documentation is essential for efficient system management and knowledge transfer.Poor or outdated documentation can hinder system maintenance and troubleshooting, making it challenging for new users or administrators to understand and work with the system.

3.1.5 Performance Bottlenecks:

- The presence of performance bottlenecks within the system leads to delays, inefficiencies, and reduced responsiveness during operation. Performance bottlenecks may arise due to limitations in hardware resources, inefficient software algorithms, or improper system configurations. Addressing performance bottlenecks is crucial for maintaining optimal system performance, enhancing user experience, and meeting operational requirements.Systems may experience performance bottlenecks, affecting their speed and responsiveness, especially when dealing with resource-intensive tasks or large datasets.

3.1.6 Limited Accessibility:

- The existing system may suffer from limited accessibility, making it challenging for users to access and interact with the system effectively. Accessibility issues could arise due to inadequate user interfaces, lack of support for assistive technologies, or restrictions on access based on physical location or device compatibility. Limited accessibility can hinder user productivity, exclude certain user groups, and lead to dissatisfaction with the system.Accessibility limitations can exclude users with disabilities, impairing their ability to use the system effectively.

3.1.7 Complex User Training:

- The complexity of the current system necessitates extensive training for users to understand its functionalities, workflows, and features. Complex user training requirements can increase the time and resources required to onboard new users, leading to productivity losses and delays in achieving proficiency. Moreover, ongoing training needs may strain organizational.

Chapter 4
PROBLEM STATEMENT AND OBJECTIVE

4.1 Problem Statement

In modern manufacturing and automation processes, the need for efficient, accurate, and adaptable object sorting systems is paramount. Traditional methods often rely on manual labor or single-purpose mechanical sorting mechanisms, which can be slow, error-prone, and inflexible. This poses a significant challenge for industries seeking to optimize their operations, reduce costs, and improve overall efficiency.

The primary issue at hand is the lack of a versatile, automated system capable of sorting objects based on their visual characteristics, particularly color. Object sorting, a fundamental task in numerous industries, stands to benefit immensely from the integration of robotic systems and computer vision. The challenge is to design and implement a vision-assisted robotic arm that can intelligently identify and sort objects by their colors in real-time, offering a more reliable and adaptable solution to the sorting problem.

By addressing this challenge, our project aims to bridge the gap between conventional sorting methods and the potential for automated, intelligent object handling. We seek to create a system that not only meets the demands of modern manufacturing, recycling, and warehousing but also paves the way for innovative applications where the fusion of robotics and computer vision can revolutionize operations and redefine efficiency.

4.2 Objectives

4.2.1 Design And Assemble a Robotic Arm:

- The primary objective is to design and assemble a robotic arm capable of performing various tasks with precision and flexibility. To design and construct a robotic arm capable of precise and coordinated movements, suitable for object manipulation and sorting tasks.

4.2.2 Implement Computer Vision:

- This It involves integrating computer vision techniques to enable the robotic arm to perceive and understand its environment, facilitating interaction with objects in real-time. To integrate a computer vision system into the project, utilizing the OpenCV library, for the purpose of capturing and processing visual information from the environment.

4.2.3 Develop Color Recognition Algorithms:

- By developing sophisticated color recognition algorithms, the robotic arm can accurately identify and differentiate objects based on their colors, enhancing its versatility in sorting tasks.To design and implement robust color recognition algorithms within the computer vision system, enabling the detection and categorization of objects based on their colors.

4.2.4 Real-time Object Sorting:

- The goal is to enable the robotic arm to autonomously sort objects in real-time based on predetermined criteria, streamlining industrial processes and increasing efficiency.To enable the robotic arm to interact with the vision system and make real-time decisions regarding the grasping and sorting of objects based on their detected colors.

4.2.5 Accuracy And Precision:

- Emphasis is placed on achieving high levels of accuracy and precision in the robotic arm's movements and actions, ensuring reliable performance in various applications. To achieve a high degree of accuracy and precision in the sorting process, minimizing errors and misclassifications.

Chapter 5

PROPOSED SYSTEM

5.1 Proposed System

Our vision-assisted robotic arm system leverages the capabilities of the Arduino Uno R3 microcontroller to automate object sorting based on color recognition.

Our proposed system is a testament to the fusion of robotics and computer vision, driven by the Arduino Uno R3, to automate and enhance object sorting by color. By utilizing this technology stack, we aim to deliver a cost-effective and versatile solution that efficiently addresses the challenges posed by manual sorting processes across various industries.

5.2 Block Diagram

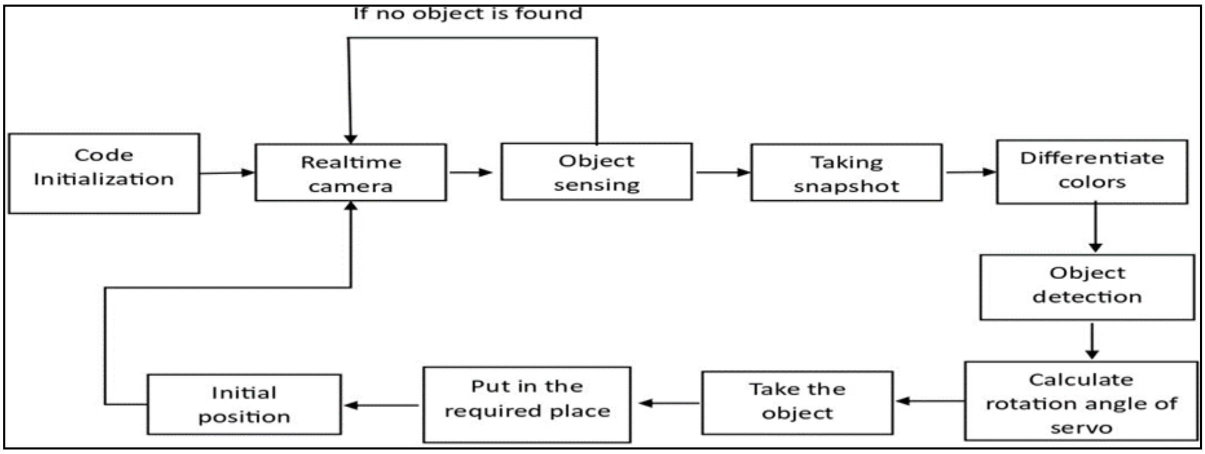

Figure 5.2.0.1: Block Diagram

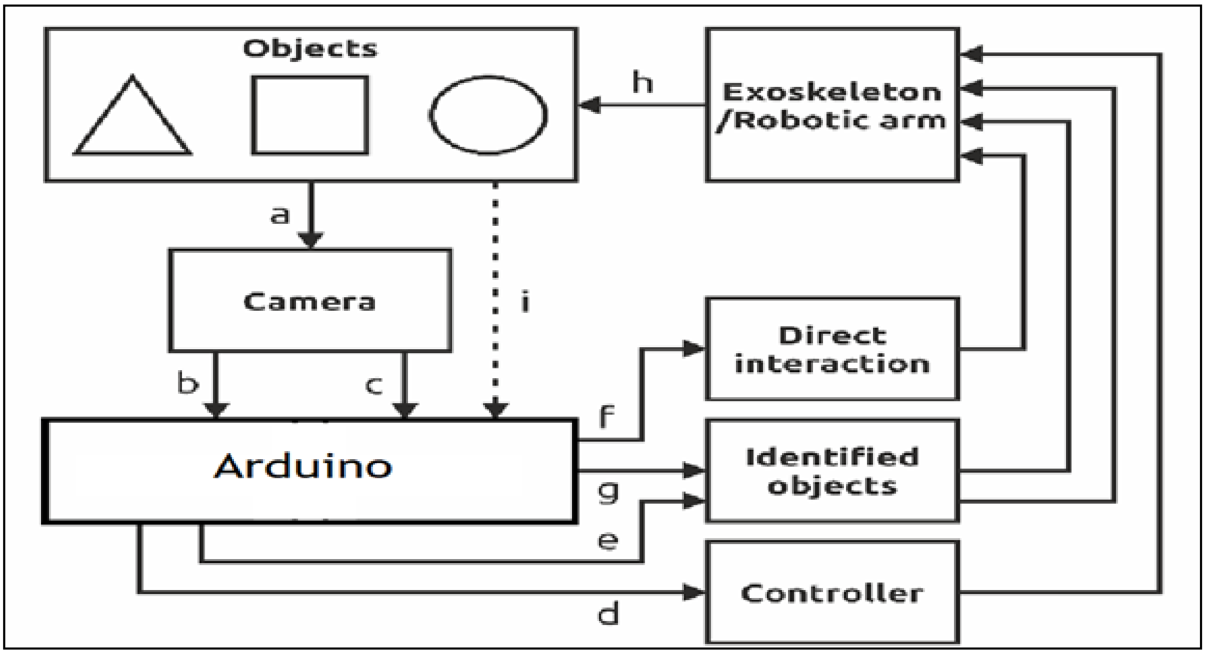

Figure 5.3.0.1: System Architecture

5.3 System Architecture

1. **Arduino Microcontroller:** At the core of the system architecture is the Arduino microcontroller, serving as the main control unit. It receives input from the camera module, processes data using onboard algorithms, and controls the movements of the robotic arm based on the detected objects.

2. **Camera Module**: The camera module captures images of the environment, providing visual input to the system. It is typically mounted on the robotic arm or positioned in the vicinity to capture the workspace. The captured images are processed for object detection and color recognition.

3. **Object Detection Algorithm**: Within the system architecture, there exists an object detection algorithm responsible for analyzing the captured images and identifying objects of interest. This algorithm employs computer vision techniques to extract features and determine the presence and location of objects within the camera's field of view.

4. **Color Recognition Module**: A color recognition module is integrated into the system to identify the colors of detected objects. This module analyzes pixel values within the detected objects' regions to determine their respective colors. It provides crucial information for sorting objects based on predefined color categories.

5.4 UML Diagrams

5.4.1 Use Case Diagram

A Use Case Diagram illustrates the interactions between actors (users) and the system (robotic arm and vision system) in terms of specific actions or use cases. Here's a basic Use Case Diagram for your project: In this diagram:

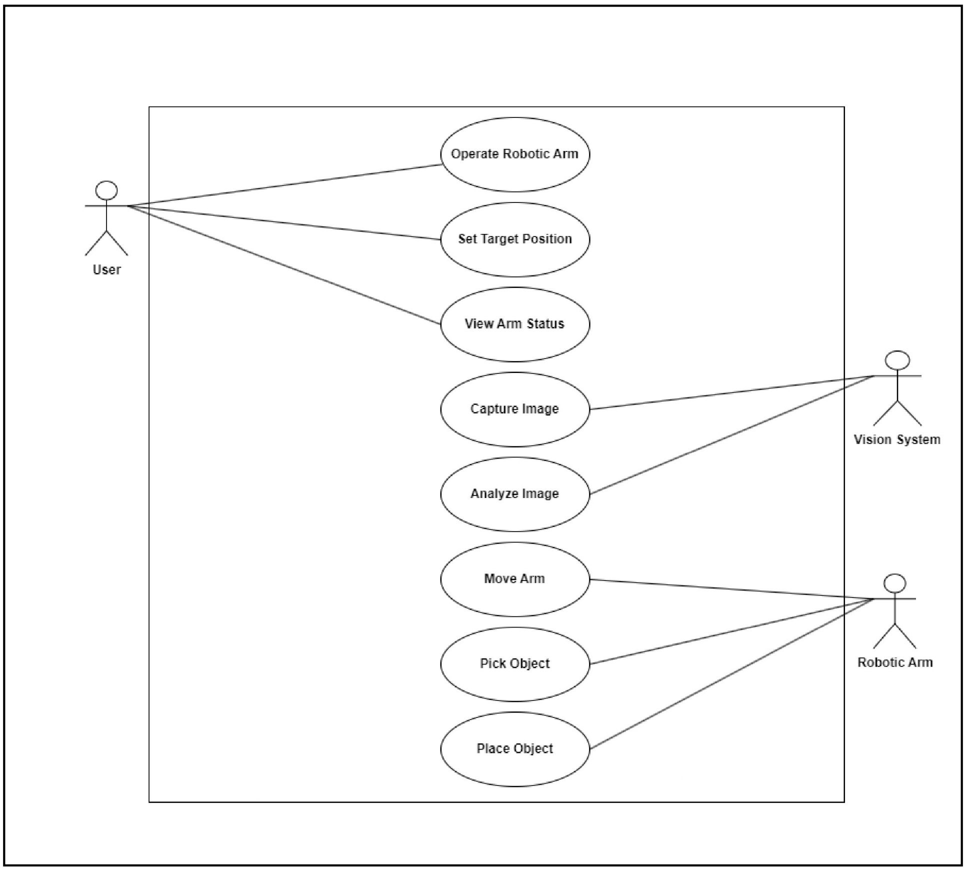

Figure 5.4.1.0: Use Case Diagram

- **User**: Represents the human operator or user interacting with the system.
- **Robotic Arm**: Represents the physical robotic arm component of the system
- **Vision System**: Not explicitly shown in the diagram, but assumed to be part of the robotic arm system, facilitating object detection and interaction.

5.4.2 Flow Chart

In this flowchart:

- The process starts at the "Start" node.
- The system activates the camera to capture images of the environment.

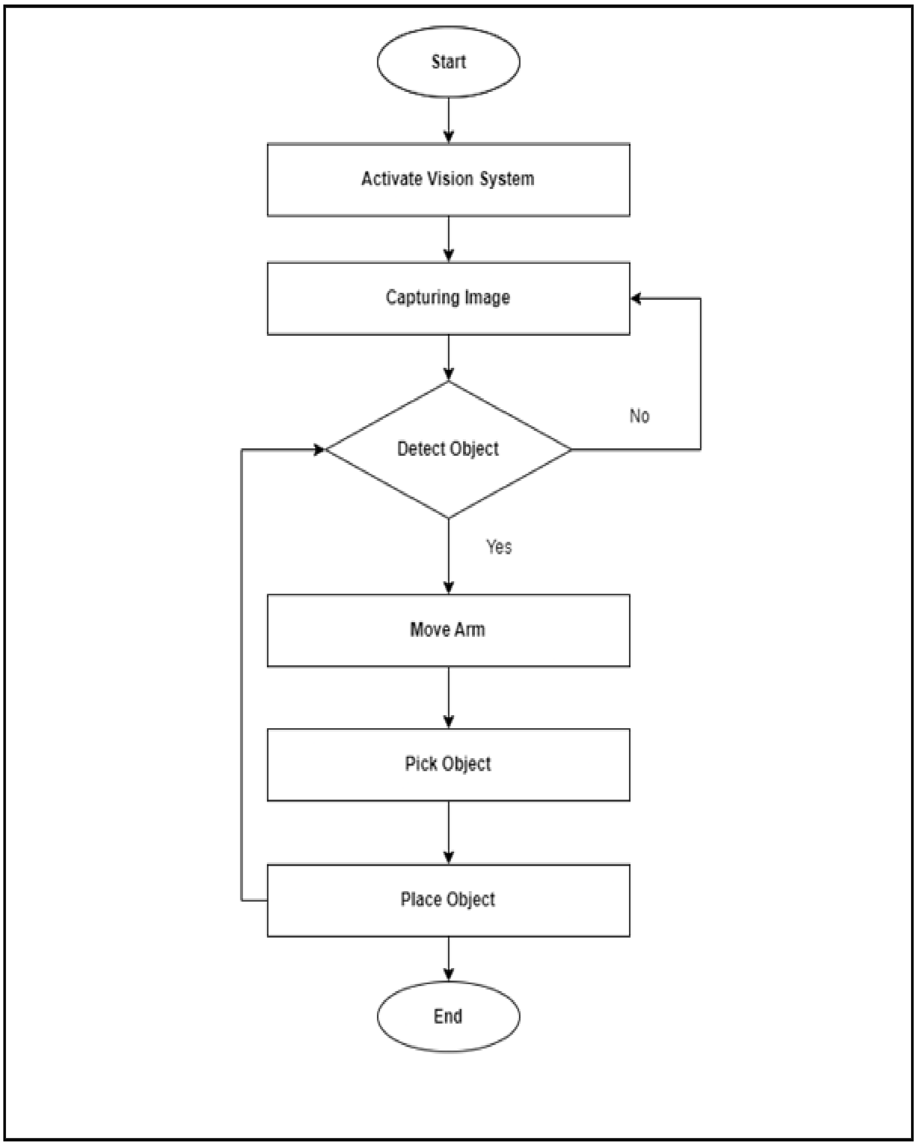

Figure 5.4.2.1: Flow chart

- Once the camera is activated, the system proceeds to detect objects within the captured images.
- After detecting objects, the system initiates the process of picking up the identified object using the robotic arm.
- Once the object is successfully picked up, the system proceeds to place the object at the desired location.
- Finally, the process ends at the "End" node.

5.4.3 Activity Diagram

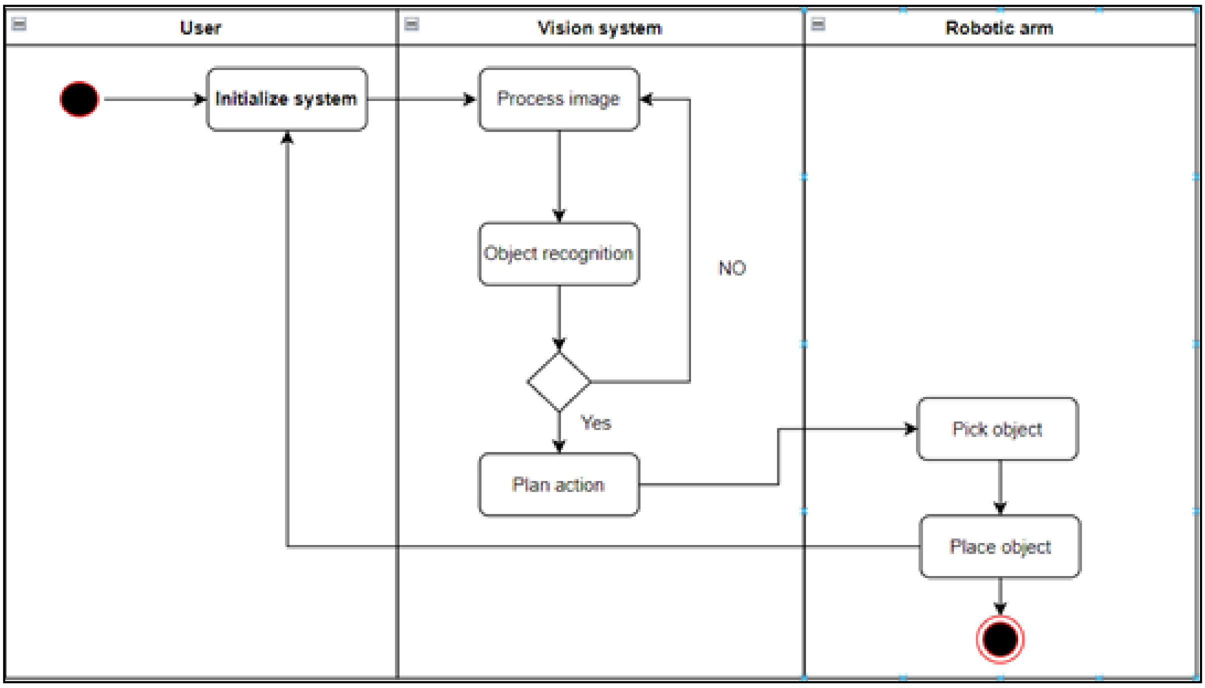

Figure 5.4.3.1: Activity Diagram

In this activity diagram:

- The process starts with the user.
- The user activates the camera.
- The vision system detects objects
- The robotic arm picks up the detected object
- The robotic arm places the object.
- Finally, the process ends.

5.5 Circuit Diagram

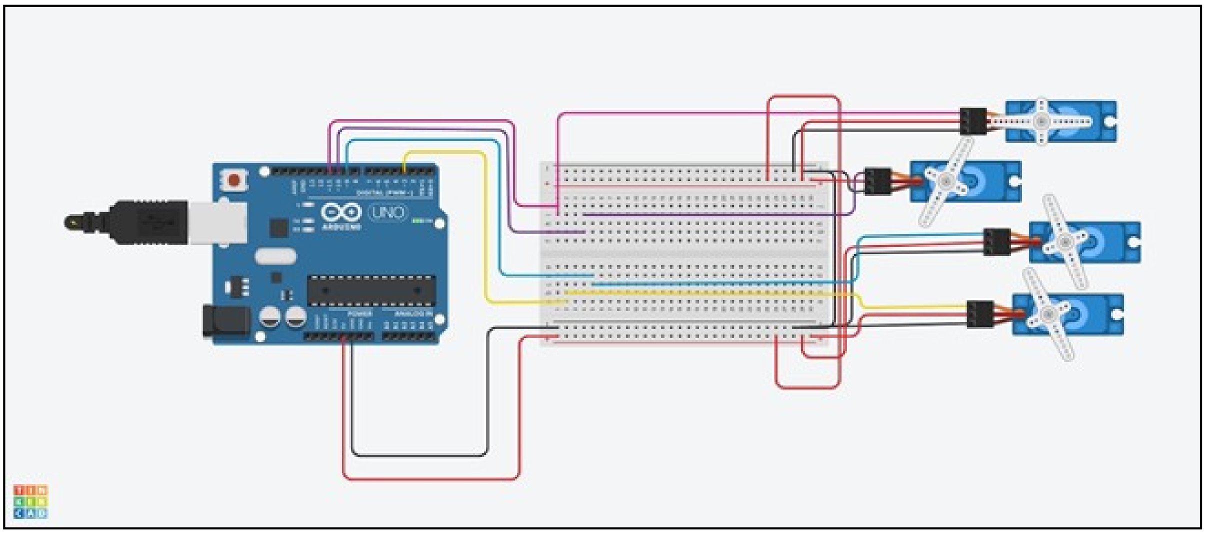

Figure 5.5.0.1: Circuit Diagram

The circuit diagram depicts the setup involving an Arduino microcontroller, a breadboard, and four servo motors. The Arduino serves as the central control unit, interfaced with the servo motors through the breadboard. Each servo motor is connected to the Arduino's digital output pins, while the breadboard facilitates easy wiring and organization of connections. The servo motors are powered through an external power source, typically a battery or an external power supply, with the ground and power connections routed through the breadboard. Control signals from the Arduino regulate the position and movement of the servo motors, allowing precise manipulation of the robotic arm's joints and end effector. This circuit forms the foundation for the robotic arm's control system, enabling coordinated movement and interaction with the environment based on programmed instructions executed by the Arduino.

Chapter 6
DESIGN DETAILS

6.1 Design Considerations

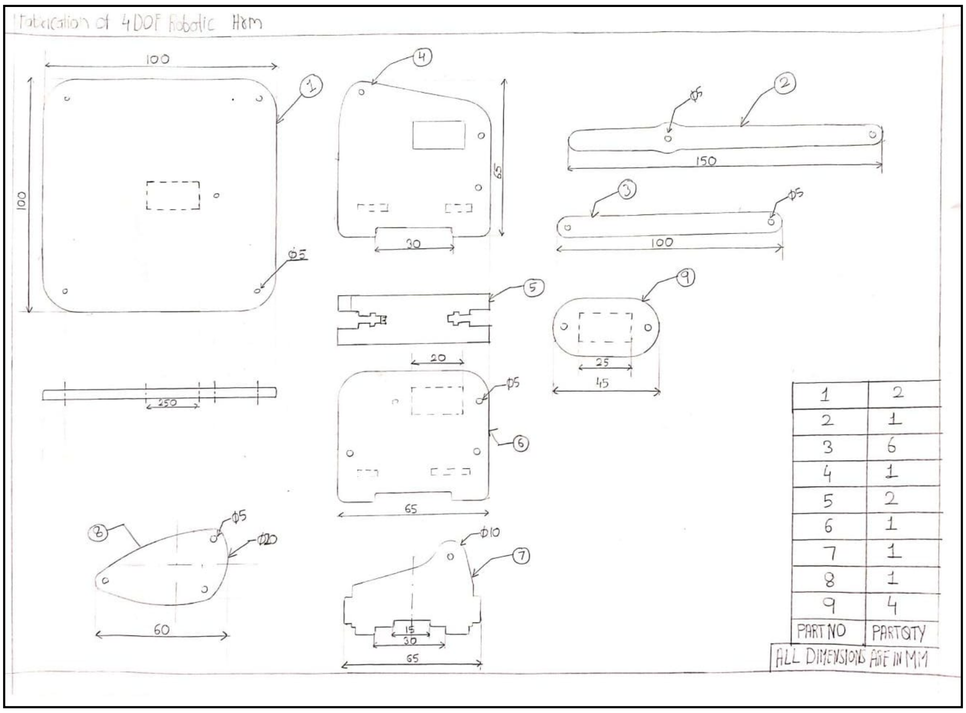

Figure 6.1.0.1: Arm Drawing

The design incorporates an Arduino microcontroller as the central processing unit, interfacing with a breadboard and four servo motors. Each servo motor is connected to the Arduino's digital output pins through the breadboard. Power and ground connections for the servo motors are routed through the breadboard, with an external power source supplying the necessary voltage. Control signals from the Arduino regulate the position and movement of the servo motors, enabling precise manipulation of the robotic arm. This setup forms the core of the robotic arm's control system, facilitating coordinated movement and interaction with the environment based on programmed instructions executed by the Arduino.

6.2 Fundamental Formulas

1. Torque Calculation: Torque (T) can be calculated using the formula (T = Fr), where F represents the force applied perpendicular to the point of rotation, and r denotes the distance from the point of rotation to the line of action of the force.

2. Forward Kinematics: Forward kinematics determines the end-effector position and orientation based on the joint angles of the robotic arm. It can be expressed as a transformation matrix T obtained through successive matrix multiplications representing the rotations and translations of each joint.

3. Inverse Kinematics: Inverse kinematics solves for the joint angles required to a T = Re a desired end-effector position and orientation. It involves the use of geometric and trigonometric relationships to determine the joint angles corresponding to a given end-effector pose.

4. Torque-Force Calculation of Model T-F calculation of Gripper Motor-D: Total weight for gripper =145 gram F1 = m1* a1 N = 0.145 * 9.8 = 1.42 N . . . . (1)

   T1 = F1 * L1 = 1.42 N * 0.01m = 0.14 kgcm . . . . . . (2)

   T-F calculation of Motor A, B, C: Total weight for Motor A, B, C = 250 gram F2 = 0.25 * 9.8 = 2.45 N. . . . . (3)

   T2 = 2.45 N * 70m = 1.75 kgcm. . . . . (4)

6.3 Experimental Result

Motor - Axis	Motion	Mass (kgcm)	Length (mm)	Torque (kgcm)	Force (N)
A-Y	Base rotation	0.250	15	0.36	2.45
B-Z	Up-Down	0.250	25	0.62	2.45
C-X	Forward & backward	0.250	70	1.75	2.45
D-Linear	Gripper Opening	0.145	10	0.14	1.42

Figure 6.3.0.1: Result Table

Chapter 7

EXPERIMENTAL SETUP

7.1 Experimental Setup

7.1.1 Performance Evaluation

Object Detection Accuracy:

- The experimental setup assesses the system's ability to accurately detect and recognize objects within its field of view. This evaluation involves comparing the system's detections against ground truth data to determine the precision, recall, and overall accuracy of object detection algorithms employed.To measure object detection accuracy, the system's ability to recognize and categorize objects based on their colors was assessed.A percentage of correctly recognized objects over a set of test objects was used as the primary metric.

Sorting Accuracy:

- The experimental setup measures the precision and effectiveness of the system in accurately sorting objects based on predefined criteria. This evaluation involves analyzing the percentage of correctly sorted objects compared to the total number of objects processed, providing insights into the system's sorting capabilities and performance.Sorting accuracy evaluated the system's capability to correctly sort objects into predefined bins based on their colors.It was measured as the percentage of correctly sorted objects relative to the total objects processed.

Processing Speed:

- This aspect of the experimental setup evaluates the system's processing speed, including the time taken to capture, analyze, and respond to input data. By benchmarking the processing speed under various conditions and workloads, researchers can identify potential bottlenecks and optimize system performance for real-time applications.Processing speed was assessed in milliseconds per object.The processing speed metric quantified the time it took for the system to recognize an object's color and make a sorting decision.

Positioning Precision:

- The experimental setup examines the precision of the robotic arm's positioning during manipulation tasks. Precision is assessed by measuring deviations from target positions and analyzing the repeatability of movements across multiple trials. This evaluation ensures that the robotic arm can perform tasks with the required level of accuracy and reliability.It was quantified by measuring the distance error between the desired and actual object positions during sorting.Positioning precision gauged the system's ability to precisely position the end- effector when manipulating objects.

Energy Efficiency:

- This aspect of the experimental setup evaluates the system's energy consumption during operation. By monitoring power usage under different workload scenarios, researchers can assess the system's efficiency and identify opportunities for optimization to reduce energy consumption and prolong battery life, where applicable. Energy efficiency was assessed by measuring the system's power consumption during continuous operation.

Continuous Operation:

- The experimental setup assesses the system's ability to operate continuously over extended periods without significant performance degradation or reliability issues. This evaluation involves running the system under sustained workloads to identify any potential overheating, stability, or wear-related concerns that may arise during prolonged operation. The system's ability to operate continuously without significant degradation in performance or hardware fatigue was examined.

System Reliability:

- This aspect of the experimental setup evaluates the overall reliability and robustness of the system in performing its intended tasks. Reliability is assessed by subjecting the system to various stress tests, including environmental conditions, mechanical stress, and software failures, to identify weaknesses and improve system resilience.System reliability assessed the robustness and uptime of the vision-assisted robotic arm system.

Decision To Action Time:

- The experimental setup measures the time taken by the system to process input data, make decisions, and execute corresponding actions. This evaluation provides insights into the system's responsiveness and latency, which are critical for real-time applications requiring prompt decision-making and action execution.Decision to action time measured the time taken for the system to process an object's color information and execute the sorting action.

Object Handling Safety:

- This aspect of the experimental setup assesses the safety measures implemented in the system to prevent accidents or injuries during object manipulation tasks. Safety evaluations include analyzing collision detection algorithms, emergency stop mechanisms, and adherence to safety standards to ensure the protection of both operators and surrounding objects.Object handling safety evaluated the system's ability to manipulate objects without causing damage to the objects or posing safety risks

Arm Precision:

- The experimental setup focuses on evaluating the precision of the robotic arm's movements and manipulations. This assessment involves measuring deviations from desired trajectories and analyzing the accuracy of grasping, lifting, and placing objects. By quantifying arm precision, researchers can optimize control algorithms and mechanical design to enhance overall performance and reliability.Arm precision assessed the accuracy and repeatability of the robotic arm's movements, particularly in reaching and grasping objects.

7.2 Hardware Set up

7.2.1 Arduino Uno

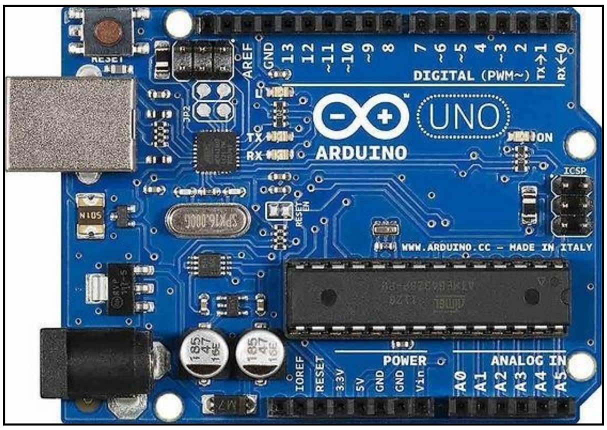

Figure 7.2.1.1: Arduino Uno

The Arduino Uno is a widely-used microcontroller board featuring an ATmega328P microcontroller with 14 digital I/O pins, 6 analog input pins, and a USB connection for programming. It operates at 5V, is open-source, and is popular for a wide range of DIY electronics and embedded systems projects.

7.2.2 Servo Motor

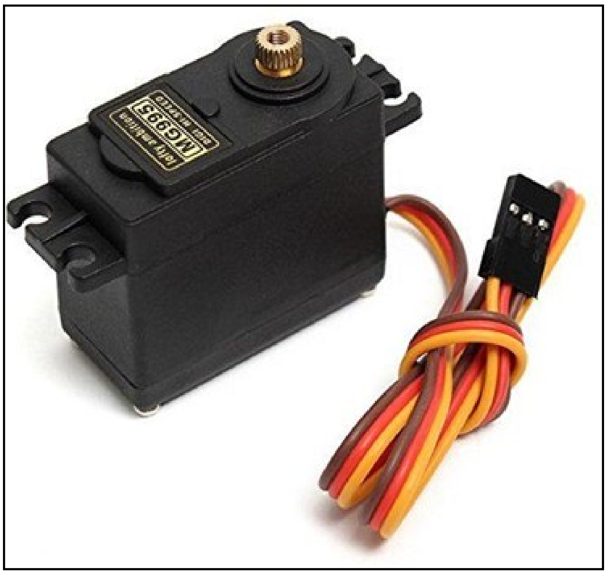

Figure 7.2.2.1: Servo Motor

A servo motor is a type of rotary actuator that is commonly used in electronics and robotics. It is designed to precisely control the angular position of the motor shaft, making it suitable for applications where accurate and controlled movement is required. Servo motors are equipped with built-in feedback systems, such as potentiometers or encoders, which provide information about the motor's current position. This feedback allows the servo motor to adjust and maintain its position in response to control signals, making it an essential component in systems like remote control devices, robotics, automation, and more. Servo motors are known for their high precision, reliability, and the ability to rotate to a specific angle, which makes them a popular choice in various motion control applications.

7.2.3 Camera

Figure 7.2.3.1: Camera

A camera module is a compact hardware component with a camera sensor and lens designed for capturing visual information. These modules are used in a variety of devices and applications, including smartphones, digital cameras, IoT devices, robots, and more, to enable image and video capture.

7.3 Software Requirements

7.3.1 Python

7.3.2 OpenCV

7.3.3 Matlab

7.3.4 Thinker CAD

7.3.5 Webot

7.3.6 AutoCAD

Chapter 8

RESULTS AND DISCUSSION

8.1 Results

The Vision Assisted Robotic Arm successfully executed color detection algorithms to accurately identify and sort objects based on their colors. Through real-time image processing, the system efficiently recognized and categorized objects, demonstrating high precision in sorting tasks. The integration of computer vision technology facilitated seamless interaction between the robotic arm and its environment, leading to efficient and reliable object sorting operations. Overall, the system showcased promising capabilities in automated object handling through color-based detection and sorting methodologies. Here's a summary of the result:

8.1.1 System Design And Integration:

Design the robotic arm with suitable actuators, sensors, and a vision system (such as a camera). Integrate the Arduino microcontroller to control the robotic arm's movements and communicate with the vision system.

8.1.2 Computer Vision Development:

Develop computer vision algorithms to process images captured by the camera and detect objects based on color. Implement algorithms to extract color information from images and classify objects into different categories for sorting.

8.1.3 Color Detection And Sorting Implementation:

Program the Arduino to receive data from the vision system and interpret color detection results. Define sorting criteria and develop control algorithms to instruct the robotic arm to pick up and place objects into appropriate bins based on their colors.

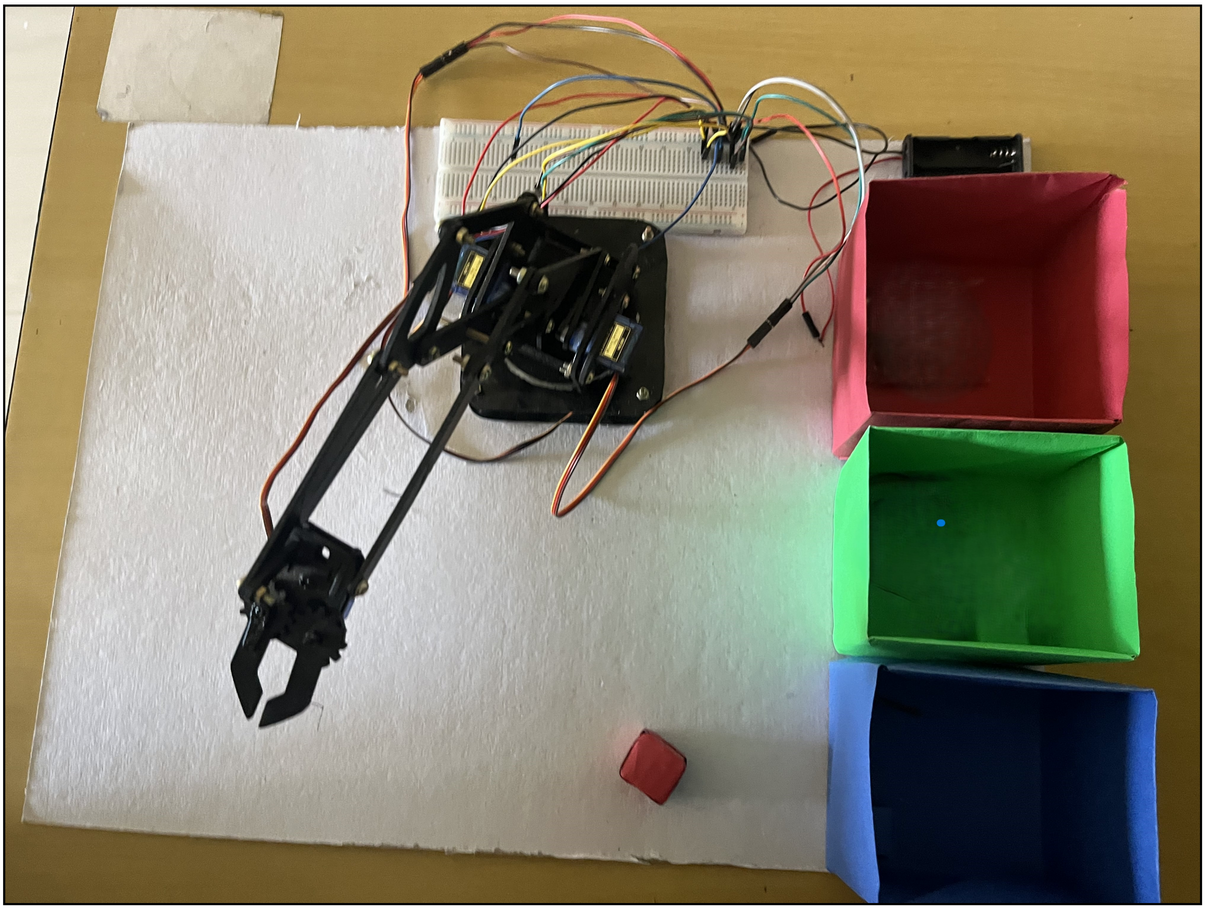
Figure 8.1.3.0: Searching the Object

- **Searching The Object**

1. **Activate Vision System:** The robotic arm starts by activating the camera or vision system to capture images of the surrounding area.
2. **Analyze Images:** Computer vision algorithms process the images to identify potential objects based on color
3. **Adjust Orientation:** As the arm identifies objects, it adjusts its position and focus based on the detected objects' locations to optimize its search strategy.

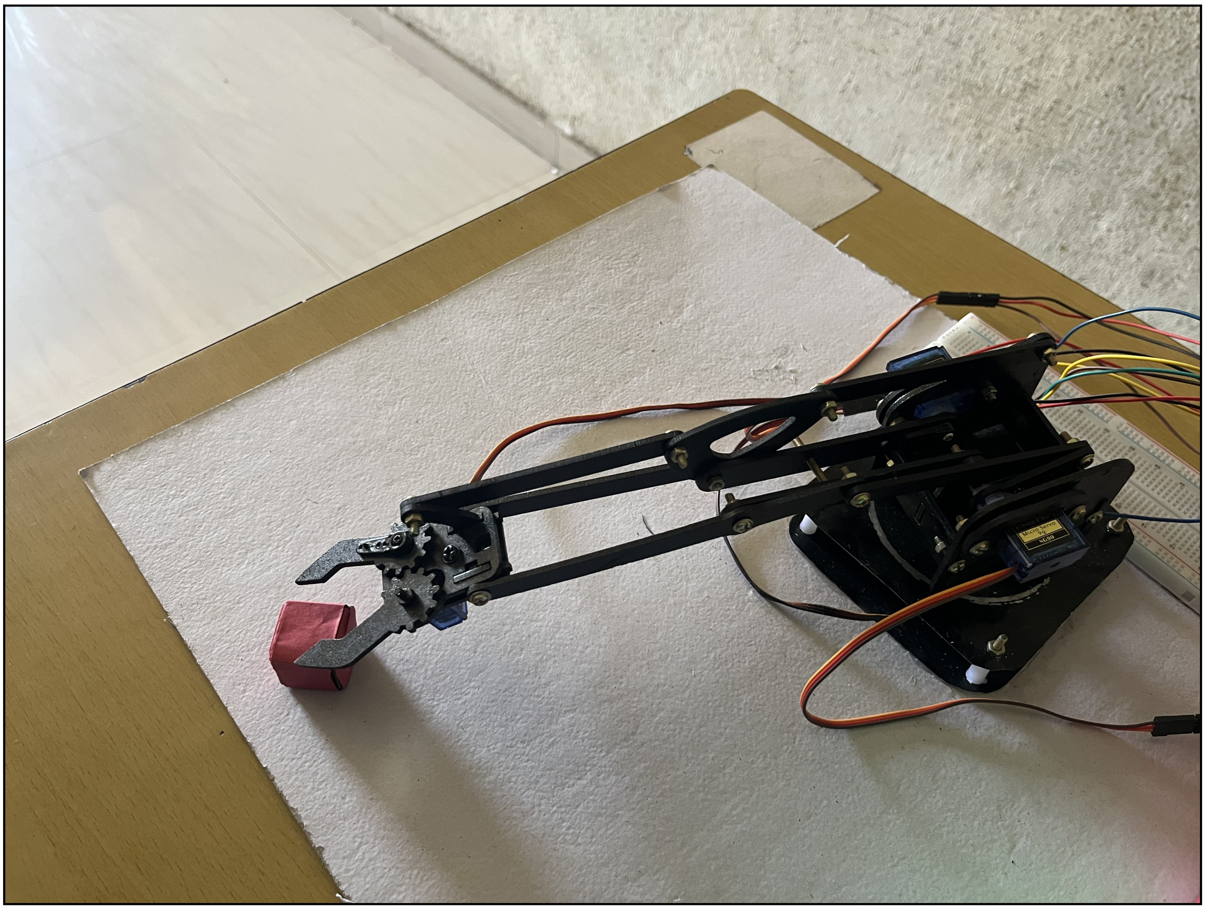

Figure 8.1.3.0: Object Detected

- **Object Detected**

1. **Identify Object:**The vision system detects objects within the camera's field of view based on predefined criteria such as color
2. **Generate Signal:**Once an object is identified, the system sends a signal to the robotic arm's control unit to indicate the object's presence and location
3. **Trigger Next Action:** The arm is prompted to proceed towards the detected object for further interaction, such as picking it up or manipulating it according to the task's requirements.

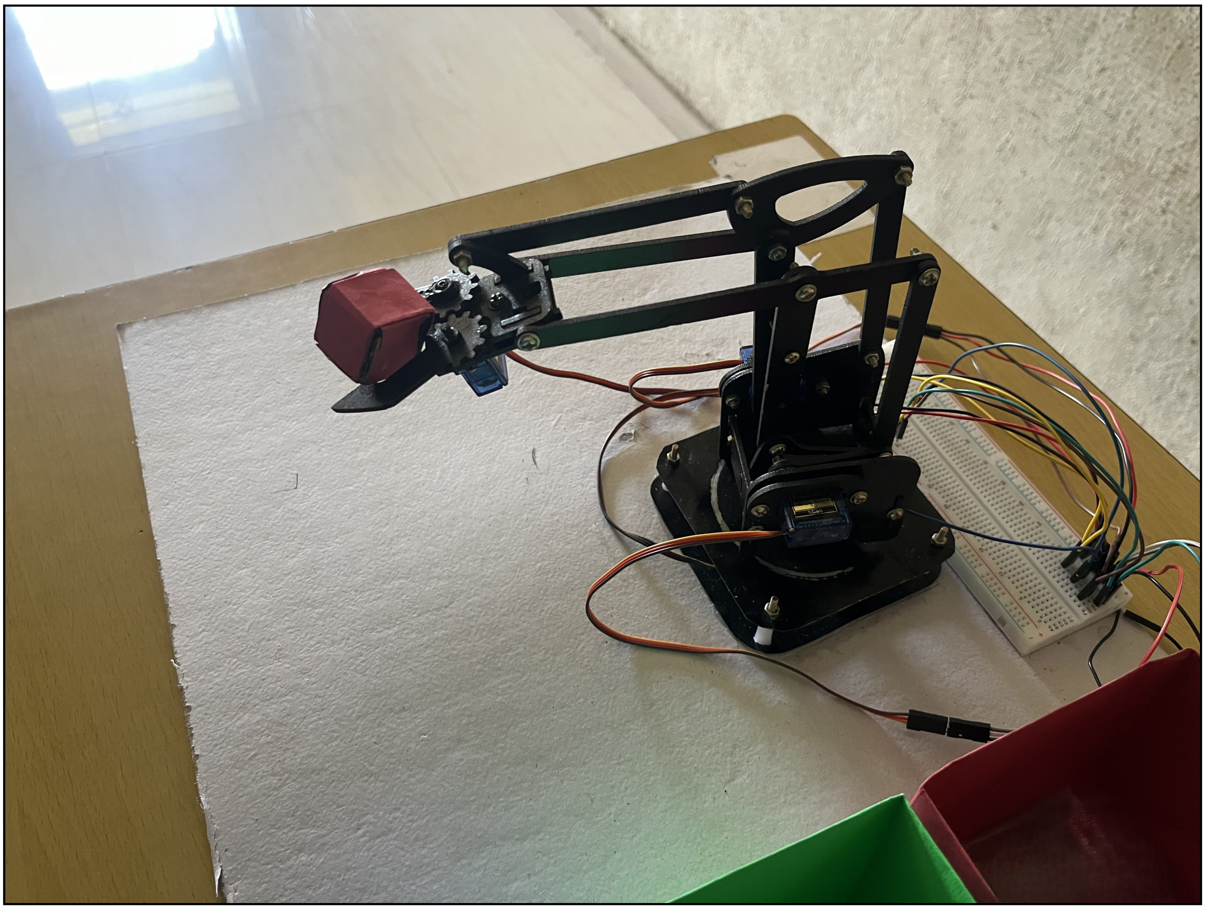
Figure 8.1.3.0: Picking The Object

- **Picking The Object**

1. **Calculate Approach:**After object detection, the robotic arm's control system calculates the movements needed to reach the object's location precisely.
2. **Activate Gripper:**The arm activates its gripper or end effector, adjusting its grip to securely grasp the object based on its size, shape, and weight.
3. **Lift Object:**Once the grip is secure, the arm lifts the object from its position using actuators or motors, preparing it for further manipulation or placement.

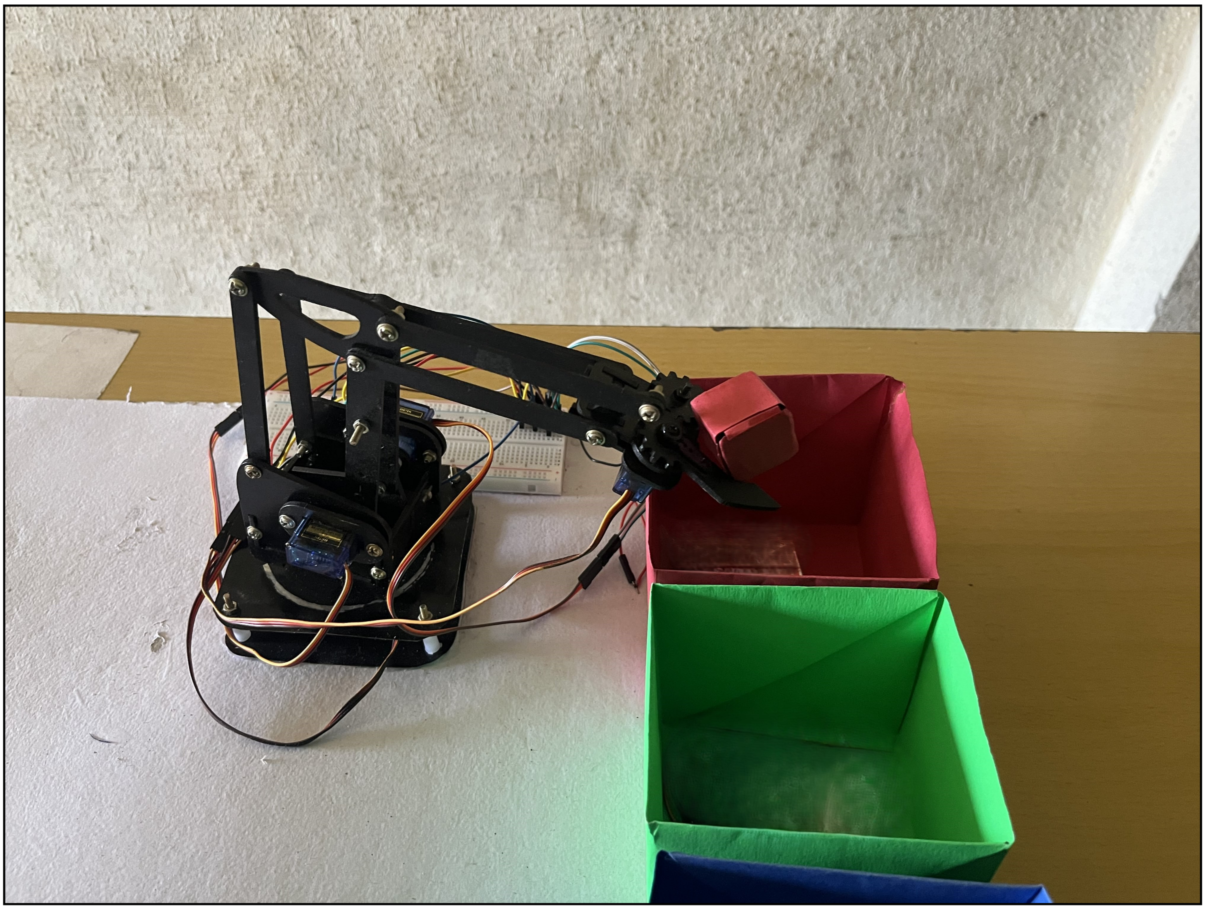

Figure 8.1.3.0: Placing the Object

- **Placing The Object**

1. **Calculate Placement:**After picking up the object, the robotic arm calculates the movements and path needed to reach the designated placement location.
2. **Adjust Position:**The arm fine-tunes its orientation and position to align the gripper with the target placement spot, ensuring the object will be placed accurately.
3. **Release Object:**Once in the correct position, the arm gently releases its grip on the object, allowing it to rest securely at the intended location, completing the process of placing the object.

Chapter 9

CONCLUSION AND FUTURE SCOPE

9.1 Conclusion

In conclusion, the fusion of vision technology and Arduino-driven robotics has paved the way for intelligent and adaptable automation solutions. Vision-assisted robot arms hold the promise of transforming industries and revolutionizing how we interact with and harness the power of robotics in our daily lives. Within the manufacturing sector, these vision-assisted robotic arms are poised to optimize production lines, streamline assembly processes, and deliver consistent quality while reducing the risk of errors. The potential for cost savings and enhanced productivity is nothing short of revolutionary. Similarly, in the field of healthcare, these technologies can revolutionize surgical procedures, making minimally invasive surgeries more precise and accessible. This promises improved patient outcomes and shorter recovery times. The future is bright, and as we stand at the crossroads of this transformative era, we must embark on this journey with a sense of responsibility, foresight, and unbridled curiosity. For the fusion of vision technology and Arduino-driven robotics has the power to shape a world that is not just automated but truly intelligent, adaptive, and ever ready to serve the needs of humanity.

9.2 Future Scope

The vision-assisted robotic arm system offers a promising future. We anticipate integrating advanced sensors, AI, and machine learning to enhance object recognition and sorting precision. Multi-object sorting, industry-specific adaptations, and collaboration with human operators are on the horizon. Improvements in energy efficiency, safety features, and remote control capabilities are priorities. Market expansion, user interface refinements, and international collaborations will further its reach and impact. This system remains poised for dynamic growth, adapting to evolving technologies and market demands across various sectors.

The fusion of vision technology and Arduino-driven robotics holds immense promise for the future. In the forthcoming years, we anticipate a deeper integration of these technologies into Industry 4.0, with fully autonomous manufacturing systems. Healthcare will benefit from vision-assisted robotic surgery, remote monitoring, and telemedicine, increasing precision and accessibility. In agriculture, smart farms equipped with vision systems will optimize crop management, and autonomous vehicles will revolutionize transportation. Personal robotics, including smart appliances and assistive devices, will become commonplace. Ethical considerations and robust regulations will be essential, and interdisciplinary collaborations will drive progress. Continuous research and innovation will remain pivotal, ensuring that the potential of these technologies is harnessed to shape a more intelligent and adaptive world in harmony with human needs.

REFERENCES

[1] C. KAYMAK and A. UCAR, "Implementation of object detection and recognition algorithms on a robotic arm platform using raspberry pi," in *2018 International Conference on Artificial Intelligence and Data Processing (IDAP)*, pp. 1–8, 2018.

[2] R.-S. Ricman, R. Szabo, and A. Gontean, "A comparative analysis of implementation performances for image processing applications used to control robotic arms," in *2018 IEEE International Conference on Environment and Electrical Engineering and 2018 IEEE Industrial and Commercial Power Systems Europe (EEEIC / ICPS Europe)*, pp. 1–6, 2018.

[3] S. F. Noshahi, A. Farooq, M. Irfan, T. Ansar, and N. Chumuang, "Design and fabrication of an affordable scara 4-dof robotic manipulator for pick and place objects," in *2019 14th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP)*, pp. 1–5, 2019.

[4] W. Zhang, C. Zhang, C. Li, and H. Zhang, "Object color recognition and sorting robot based on opencv and machine vision," in *2020 IEEE 11th International Conference on Mechanical and Intelligent Manufacturing Technologies (ICMIMT)*, pp. 125–129, 2020.

[5] G. Alnowaini, A. Alttal, and A. Alhaj, "Design and simulation robotic arm with computer vision for inspection process," in *2021 International Conference of Technology, Science and Administration (ICTSA)*, pp. 1–6, 2021.

[6] D. H. Won, Y. J. Lee, S. Sung, and T. Kang, "Integration of vision based slam and nonlinear filter for simple mobile robot navigation," in *2008 IEEE National Aerospace and Electronics Conference*, pp. 373–378, 2008.

[7] X.-x. Qi, B.-x. Ma, and W.-d. Xiao, "On-line detection of hami big jujubes' size and shape based on machine vision," in *2011 International Conference on Computer Distributed Control and Intelligent Environmental Monitoring*, pp. 96–102, 2011.

[8] S. Takahashi and B. Ghosh, "Motion and shape identification with vision and range," *IEEE Transactions on Automatic Control*, vol. 47, no. 8, pp. 1392–1396, 2002.

[9] R. Reddy and S. R. Nagaraja, “Integration of robotic arm with vision system,” in *2014 IEEE International Conference on Computational Intelligence and Computing Research*, pp. 1–5, 2014.

Chapter 10

CERTIFICATES

ISSN No. : 2321-9653

iJRASET

International Journal for Research in Applied Science & Engineering Technology

IJRASET is indexed with Crossref for DOI-DOI : 10.22214

Website : www.ijraset.com, E-mail : ijraset@gmail.com

ISRA Journal Impact Factor: 7.429

45.98 INDEX COPERNICUS

THOMSON REUTERS

10.22214/IJRASET

TOGETHER WE REACH THE GOAL

SJIF 7.429

Certificate

It is here by certified that the paper ID : IJRASET59740, entitled

Vision Assisted Robotic Arm Using Arduino

by

Mr. Khan Mohammad Tahir

after review is found suitable and has been published in

Volume 12, Issue IV, April 2024

in

International Journal for Research in Applied Science & Engineering Technology

(International Peer Reviewed and Refereed Journal)

Good luck for your future endeavors

Editor in Chief, iJRASET

Figure 10.0.0.1: Certificate Of Mr.Khan Mohammad Tahir

Figure 10.0.0.2: Certificate Of Mr.Mohammed Yusuf

Figure 10.0.0.3: Certificate Of Mr.Chaudhary Shahbaz

Figure 10.0.0.4: Certificate Of Mr.Bakhed Avesh

Figure 10.0.0.5: Certificate Of Prof. Monika Pathare