International Journal of Advanced Trends in Engineering and Management
(ISSN 2583-7052)

pp. 57-63, May

AI-Powered Robotic Arm for Color-Based Object Sorting and Voice Command Execution: A Comprehensive Analysis

Dr. B. Shadaksharappa[1], Aditya singh[2], Parul mishra[3], Pulasa teja sree[4]

[1] Principal, Sri Sairam College of Engineering, Bangalore, India
[2, 3, 4] Students, Computer Science Engineering, Sri Sairam College of Engineering Bangalore, India

ABSTRACT: This paper provides an overview of the design and development of an automated robotic arm system for sorting objects by color detection or through voice commands, which combine computer vision and speech recognition. The robotic arm is controlled by an Arduino microcontroller that commands servo motors to perform the robotic pick-and-place, while object detection using Python with the OpenCV library and numerical computing using NumPy. An external camera, connected to the computer, observes real-time videos to detect color (e.g., red versus blue), and a voice recognition module that allows users to verbally give commands for the robotic arm to sort objects or colors. When the robotic arm observes a specific color or voice command, the robotic arm is initiated to pick-and-place the object into an area designated by the user. The overall result of this project showcases an efficient, cost-effective, and interactive system for automated sorting applications in industrial and home areas.

Keywords: Robotic Arm, Arduino, OpenCV, Color Detection, Voice Control, Computer Vision, Speech Recognition, Automation, Sorting.

1. Introduction

Automation has also become crucial across sectors to make processes more efficient and minimize the chances of errors due to humans in repetitive functions like object sorting. Manual traditional sorting processes consume time, take labor, and are mostly unreliable. To correct these shortcomings, this project showcases a low-budget robotic arm device that incorporates computer vision and voice command for sorting objects automatically.

The system uses an external camera to identify objects based on color through OpenCV and NumPy libraries in Python. Voice automation is also integrated through a speech recognition module, enabling users to command the robotic arm verbally. The Arduino microcontroller manages servo motors that facilitate the pick-and-place functionality of the robotic arm.

Through the integration of color recognition and voice control, the system presented herein presents a flexible, interactive, and effective solution that can be applied in industry and at home. Such integration shows that low-cost hardware and open-source software can be used to develop smart robotic systems with real-time decision-making ability and ease of use.

2. Literature Survey

2.1 Robot Arm System Design and Development for Sorting and Classification by Machine Vision (2022)

This article by Cong et al. describes a double robotic arm system that classifies objects according to shape and size with the aid of machine vision. The system utilizes a master-slave hierarchical control system in which an embedded computer is responsible for image processing while microcontrollers individually control each robotic arm. Object recognition is done through algorithms such as contour detection, centroid computation, and shape thresholding. Smooth robotic arm movements are also realized through the application of trapezoidal motion profiles in order to limit vibration during sorting operations. Experiment results validate that the system increases production line efficiency by decreasing sorting time and accuracy, making the system appropriate for industrial automation contexts.

2.2 Development of a Color-Based Image Recognition System for Robotic Sorting and Picking (2024)

In this study, Sidehabi et al. create an automatic color-sorting robotic system based on color recognition through the HSV color model. The system uses a Dobot Magician robotic arm, an Arduino Mega, a photoelectric sensor, and a conveyor belt for handling materials. Image processing is carried out in Python to classify objects according to color and control sorting activities. Throughout a series of trials, the system attained 100% color class accuracy, presenting high reliability. The method displays a drastic enhancement over traditional sorting by hand and presents a streamlined and adaptive solution for automation using color-based manufacturing through open-source hardware and computer vision tools

2.3 Real-Time Image Processing and Robot Gripper Mechanism Control-Based Object Sorting System (2023)

Sanwar and Ahmed put forth a system that sorts objects on the basis of shape as well as color using real-time image processing. Images are captured using a webcam, and object features are detected by Python-based algorithms. A robotic gripper is controlled using an Arduino UNO microcontroller actuated by servo motors for picking and placing objects precisely on a conveyor system. Objects are categorized into various angular zones according to detection outcomes. This system emphasizes the viability of low-cost, adaptable robotic automation in industrial applications. It minimizes manpower, enhances operational effectiveness, and demonstrateshow cheap technologies can perform moderately complicated object sorting applications in real-time.

2.4 Vision-Assisted Robotic Arm Using Arduino (2024)

This project involves a low-cost robotic arm driven by Arduino and directed by real-time computer vision through OpenCV. The system detects the color and shape of objects detected through a live camera feed and performs pick-and-place operations through servo motors. The robotic arm reacts dynamically according to the results of image processing. This model shows the viability of coupling vision systems with simple robotic arms for educational and small-scale industrial automation applications. It focuses on real-time decision- making and offers a scalable platform for future enhancements such as the inclusion of voice control or the extension to multi-object detection platforms.

2.5 MaxArm: Arduino Robot Arm Conduct Color Sorting in ways

MaxArm provides a two-method robotic sorting system, where objects can be sorted either by OpenCV-based vision detection or by direct color sensors. The robotic arm controlled by Arduino utilizes servo motors to pick and place objects according to color detected. This hybrid method provides flexibility, making it suitable for various resource environments — sensor-based sorting for rapid tasks and camera-based detection for high-precision requirements. The project effectively demonstrates how the combination of vision and sensor-based feedback can improve system flexibility, making it applicable to dynamic environments where the requirements vary and varying levels of accuracy and speed are required for sorting tasks.

2.6 Design of a Color-Based Image Recognition System for Robotic Sorting and Picking (2024)

Wetenriajeng et al. created a color-sorting robot system with a Dobot Magician arm integrated with Arduino and Python programming. Through HSV color space usage, the system classifies and sorts items with superior accuracy. It supports a conveyor belt and photoelectric sensor for detecting and aligning objects prior to sorting. Real-time processing allows the robot to pick and place the objects in a highly efficient manner into categorized bins. Their experimental results show consistent high performance, highlighting how combining robotics, image processing, and simple sensor integration can significantly enhance object handling and streamline manufacturing and assembly processes in automated production lines.

3. Methodology

The objective of the project is to develop an automated robotic arm to sort objects based on color utilizing a combination of OpenCV, Python, Arduino, and servo motors. The system is split into three main sections: image processing, robot control, and user interface. OpenCV is utilized to capture and process the live camera feed, detecting objects based on color via HSV filtering. The robotic arm is powered by an Arduino microcontroller, which controls the movement of the arm depending on instructions from Python through serial communication. The robotic arm also comes with voice command capability, and users can drive the robotic arm by giving it commands such as "pick up the blue box," interpreted through Python's speech recognition library. The servo motors drive the robotic arm to grasp objects and drop them in the color-coded boxes. Hardware-software integration is verified by extensive testing to provide accurate color recognition, smooth movement, and efficient voice command operation.

4. Proposed System

The system proposed is an automated robotic arm that can identify and sort objects by color. It uses OpenCV for real-time image processing, where the environment is captured by a camera, and objects are identified according to their color characteristics. The images captured are processed through color filters in the HSV (Hue, Saturation, Value) color space, enabling precise identification of various colors such as blue, black, red, etc.The servo motors controlled by an Arduino microcontroller are placed within the robotic arm, programmed in C++. The arm has a precise motion to collect objects and put them into respective colored boxes as identified by the color sensor. Blue objects are placed into a blue box, and black objects into a black box.The system is further bolstered with voice command integration so users can operate the robotic arm verbally. Utilizing Python's speech recognition library, users can send commands like "pick up the blue box," and the system will perform the action without human intervention.

The architecture of the system includes a camera, processing component (Python using OpenCV and NumPy), robotic arm based on Arduino, and voice interface, which integrate with each other harmoniously. This automation not only makes sorting easier but also adds an easy-to-use interface for real-time command, enhancing efficiency and user friendliness

4.1 Design Overview

The system design of the color-based object sorting automated robotic arm combines hardware and software elements to provide smooth color detection, sorting, and user control. The following is a summary of the system design:

4.1.1 Hardware Components

Camera: A high-definition camera takes real-time video of the surroundings. The camera is placed to capture the objects to be sorted by color.

Robotic Arm: Servo motors power the robotic arm. The robotic arm is optimized for precise action, with the ability to grasp and deposit items into specified containers.

Arduino Microcontroller: The robotic arm control center is a programmed Arduino board in C++. It interfaces to the Python application through serial interface communication, exchanging commands and passing control signals back to the servo motors.

Microphone: A microphone captures voice commands, which are then processed by the speech recognition software.

4.1.2 Software Components

OpenCV: The Python library OpenCV is employed for real-time image processing. It grabs the video stream from the camera and then applies color detection algorithms to detect objects based on their HSV values.

NumPy: The NumPy library is used for efficient numerical computations, particularly in processing pixel data from the camera feed and matrix operations.

Speech Recognition: A voice command interface is implemented using Python's speech recognition library. It receives user commands such as "pick up the blue box" and interprets them as actionable tasks for the robotic arm.

4.1.3 System Workflow

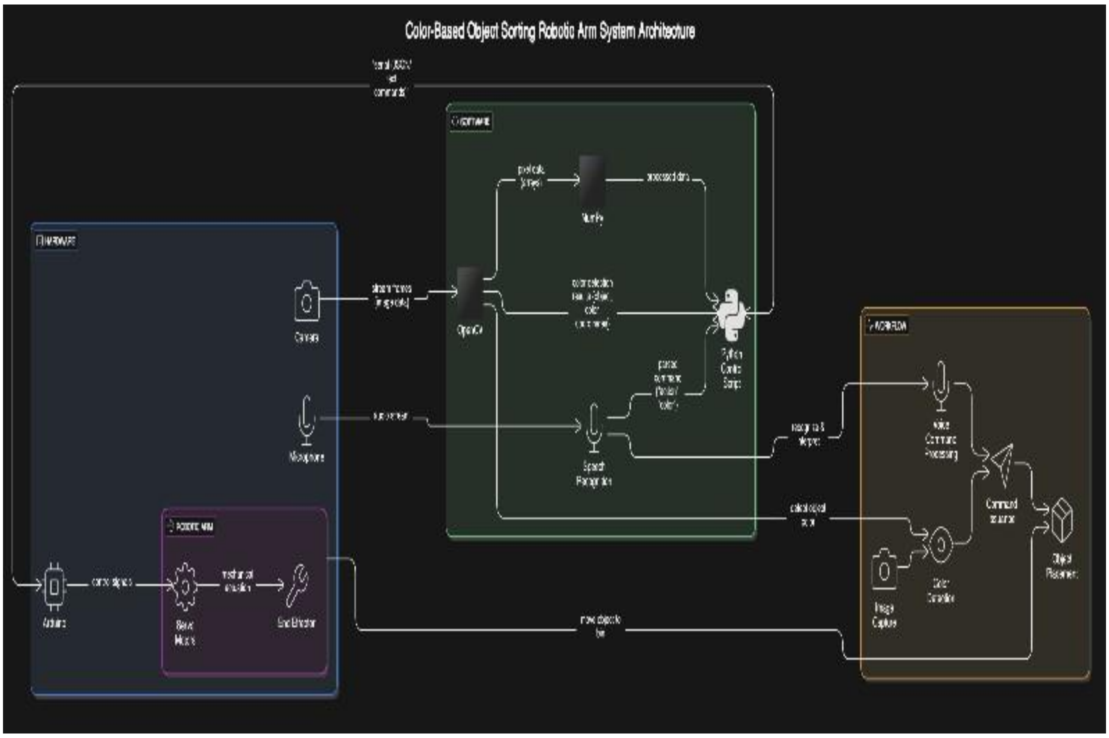
Image Capture: The camera continuously takes frames of the scene.

Color Detection: OpenCV scans each frame, detects objects by color, and decides which color object to select.

Control Command: The Python script issues a command to the Arduino, telling it to drive the robotic arm to the position to pick up the object.

Voice Command Integration: Alternatively, the system waits for voice commands. When it receives a valid voice command, it performs the associated action (e.g., picking a certain colored box).

Object Placement: The robotic arm picks up the object, moves to the right position, and places it in the correct color box.



4.2 System Architecture

The automated robotic arm color sorting system's system architecture is comprised of several crucial components that handle different activities within the whole operation.

Camera and Image Processing: The process starts with a camera recording the actual surroundings in which objects are located. The camera transfers video feed to a Python program, and this program employs OpenCV for image processing. OpenCV processes frames to detect objects by observing their colors, employing the HSV (Hue, Saturation, Value) color space to detect them well even in different lighting conditions.

Control System (Arduino): As soon as a color is sensed, the Python program issues a control signal to an Arduino microcontroller through serial communication. The Arduino, which is written in C++, decodes these signals and activates the servo motors attached to the robotic arm. The arm has been engineered to make accurate movements, and as such, can pick up and drop objects according to the detected color.

Voice Command Interface: The system includes voice commands, allowing users to command the system to perform actions like "pick up the blue box" using a microphone. The Python speech recognition module interprets these commands and initiates respective actions, sending the required signals to the Arduino.

Sorting and Placement: The robotic arm grasps objects and places them into corresponding color-coded boxes, making the sorting process automatic.

4.3 Requirements

The proposed system requires the following hardware and software components:

4.3.1 Hardware:

- **Camera:** High-definition camera for real-time object capture. Robotic Arm: Equipped with servo motors for precise movements.
- **Arduino Microcontroller:** For controlling the robotic arm and interpreting commands.
- **Microphone:** For voice command input.
- **Color-coded Boxes:** For sorting objects based on detected colors.

4.3.2 Software:

- **Python:** For controlling the system, integrating OpenCV and voice recognition libraries.
- **OpenCV:** For image processing and color detection.
- **NumPy:** For numerical computations related to pixel data processing. Arduino IDE: For programming the Arduino to control the robotic arm. Speech Recognition Library: For processing voice commands in Python.

4.4 Working

The robotic arm used for object color sorting is based on an automated robotic arm and has the function of real-time image processing, robotic control, and voice command. The process starts by the camera taking the real-time workspace where the objects are positioned for sorting.

The camera continues to feed video into the Python program, which uses OpenCV to analyze the video stream. The OpenCV library translates the captured image from RGB to HSV (Hue, Saturation, Value) color space to better detect colors despite lighting fluctuations. The system uses pre-defined color filters for particular hues (e.g., blue, black, or red), which filter out the objects according to the color detected.

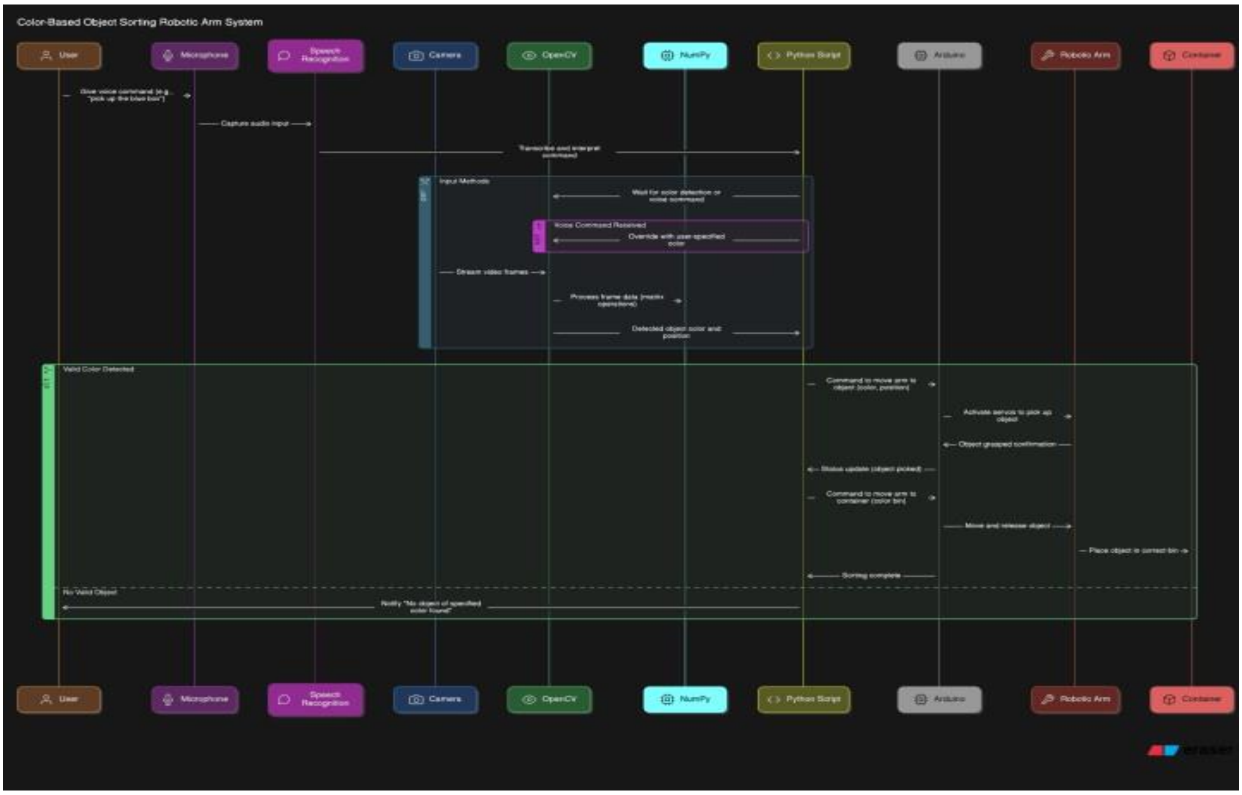
After detecting the color of an object, the program computes its centroid and sends the object's coordinates to the Arduino microcontroller. The Arduino then decodes these signals and drives the servo motors of the robotic arm, moving the arm to the object's location.

The gripper-equipped robotic arm moves to the object identified. The servo motors enable the arm to change its position and grasp the object by turning on the gripper, which holds the object firmly. Once the object is gripped, the robotic arm travels to the proper color-coded box, as determined by the system, and drops the object into the correct box.

Besides this, voice commands are also supported by the system. A microphone records user commands, and these are analyzed by the speech recognition module implemented in Python. For example, commands such as "pick up the blue box" are analyzed, and the action corresponding to it is carried out, making the robotic arm pick up the object of the corresponding color. After the completion of the task, the robotic arm goes back to its home position, prepared for the next task.

The system operates independently, constantly monitoring the workspace and responding to new items that are being placed for sorting. Following each action, feedback can be provided by the system, either in speech via the voice module or visually. In addition, should an error arise, such as an object failing to be recognized or an obstacle in the arm's path, the system will notify the user, allowing it to run smoothly.

By the integration of OpenCV, servo motors, Arduino control, and voice commands, the system effectively and independently sorts objects based on color, providing both visual and voice interaction, making it very efficient and user-friendly.



5. Conclusion

The color-based object sorting robotic arm seamlessly showcases the integration of robotic control, image processing, and voice command capabilities to solve practical object sorting challenges. By utilizing cutting-edge technologies such as OpenCV for real-time object color detection, the system allows the robotic arm to precisely detect objects based on color and sort them into classified boxes with minimal manual input. The microcontroller Arduino makes sure that there is accurate control over the servo motors so that the arm will be able to move precisely in order to take up and place objects.

Including voice command function offers a hands-free and straightforward method for the user to access the system. Through the speech recognition module, users can command the robotic arm to grab and sort objects without physically having to interact with the system. Not only does this feature make the system easier to use but also more accessible to a greater number of users, such as those with reduced mobility or technical skills.

The system's capability to process visual data in real-time, accurately control physical devices, and act on voice commands makes it extremely versatile and scalable. Such versatility enables the system to be utilized in multiple domains, such as manufacturing, logistics, warehousing, and home automation. It provides a solution to industries that are dependent on efficient and accurate sorting processes, such as inventory management or product assembly, where automation can minimize labor costs and enhance operational efficiency.

In addition, the successful integration of hardware and software demonstrates the potential of AI-powered automation in streamlining intricate processes. The system paves the way for more advanced, autonomous systems that can transform industries. With the capability to enhance productivity, minimize human error, and enhance efficiency, the project is a stepping stone toward more advanced uses of robotics and artificial intelligence in real-world applications.

References

1. Akkas U Haque; Humayun Kabir; S. C. Banik; M. T. Islam, Year: 2023, "Development of a voice controlled robotic arm".

2. Vineeth Teeda; K. Sujatha; Rakesh Mutukuru, Year: 2024, "Robot voice a voice controlled robot using Arduino".

3. Abu Salman Shaikat; Suraiya Akter; Umme Salma, Year: 2020, "Computer vision based industrial robotic arm for sorting objects by color and height", Journal of Engineering Advancements, Vol: 1, No: 04, pp. 116 – 122.

4. T. Shahoriar; M. Hossain, Year: 2020, "Computer vision-based robotic arm for object color, shape, and size detection". Journal of Robotics and Control (JRC), Vol: 1, No: 2, pp. 139 – 145.

5. Cong Duy Vo; Duy Anh Dang; Phuong Hoai Le, Year: 2021, "Development of multi-robotic arm system for sorting system using computer vision", Journal of Robotics and Control (JRC), Vol: 2, No: 1, pp. 1– 9.

6. K. M. Tahir; M. Yusuf; C. Shahbaz; B. Avesh; M. Pathare, Year: 2024, "Vision assisted robotic arm using Arduino". IJRASET.

7. U. Salma; S. Akter; A. S. Shaikat, Year: 2020, "Design and development of a robotic arm for sorting colored objects". ResearchGate.

8. Abu Salman Shaikat; Suraiya Akter; Umme Salma, Year: 2020, "Computer vision based industrial robotic arm for sorting objects by color and height", Journal of Engineering Advancements, Vol: 1, No: 04, pp. 116 – 122.