



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.2025.68784>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Multifunction Programmable Automation Software Application for Robotic Arm

Dr.NitaThakare¹, Mrunmayee Atey², Lochan Futane³, Sakshi Menkudale⁴

CT Dept., PCET, Nagpur

Abstract: *This paper explores the intersection of robotics and computer vision, focusing on developing a robotic arm capable of recognizing and interacting with objects based on their color and shape—much like human vision. The process starts with selecting or designing a robotic arm equipped with high-resolution cameras and advanced sensors. These components enable the arm to gather detailed visual data, forming the foundation of its decision-making abilities. At the core of this system is image processing. Cutting-edge algorithms analyze raw visual input, detecting colors and shapes in real time. This allows the robotic arm to respond dynamically rather than simply executing preprogrammed tasks. To bridge perception and action, sophisticated control algorithms are implemented. These guide precise movements for tasks like object sorting, pick-and-place operations, and quality control in industrial settings. The potential applications of this technology span manufacturing, logistics, agriculture, healthcare, and education. By seamlessly integrating robotics with computer vision, this paper paves the way for intelligent automation and enhanced human-robot collaboration.*

Keywords: *Color and Shape Detection AI, Artificial Intelligence, Robotics, Computer Vision, Robotic Arm, Image Processing*

I. INTRODUCTION

In the world of computer vision, object detection and tracking are fundamental capabilities, underpinning applications across numerous industries. At its core, this process involves analyzing image sequences to detect, classify, and monitor objects over time. While humans find object recognition effortless, training machines to achieve similar levels of accuracy presents a formidable challenge. Techniques in object recognition range from pattern matching and feature-based learning to deep learning-driven approaches, with shape detection playing a pivotal role in applications such as robotics, medical imaging, and assistive technologies.

As growing trends in object sorting lean toward robotic automation over manual methods, particularly because human color perception is prone to inconsistency, leading to inefficiencies over time. By integrating automation, tasks can be performed with greater accuracy and without the need for continuous human oversight.

This paper presents an intelligent robotic system specifically designed for industrial applications, capable of detecting and sorting objects based on their distinct shapes and colors. The system employs HSV-based color detection and contour-based shape recognition techniques, using the OpenCV algorithm for shape approximation.

A. Significance of Color and Shape Detection in Robotics

Color and shape detection are essential aspects of object recognition in robotics. By enabling a robotic arm to detect color, it can sort, categorize, or prioritize objects based on their visual attributes. Similarly, shape detection allows the robotic system to recognize and manipulate objects of different geometries, facilitating tasks such as picking, placing, and sorting items based on both physical form and appearance.

For instance, in the packaging industry, a robotic arm equipped with color and shape detection capabilities can distinguish between different products, ensuring accurate sorting or packaging. In electronics manufacturing, such a system can differentiate between various components based on their color-coded labels or shapes, leading to increased efficiency and reduced errors in assembly processes. Color detection is achieved through the use of color sensors, typically employing photodetectors that respond to different wavelengths of light. The software processes the data from these sensors and determines the color of the object in real time. Shape detection, on the other hand, is often carried out using machine vision systems, where a camera captures an image of the object, and the software applies algorithms to analyze the object's contours, edges, and dimensions.

B. Multifunction Programmable Automation

The core of this paper lies in the development of a multifunction programmable automation software application that integrates both color and shape detection capabilities.

This software is designed to allow the robotic arm to autonomously handle a variety of objects based on specific user-defined criteria. By leveraging a graphical user interface (GUI), the software enables operators to program the robot's tasks without needing in-depth coding knowledge. Through this interface, users can set parameters such as the color or shape of objects that need to be manipulated, ensuring the robotic arm performs tasks accurately and efficiently.

The multifunctionality of the software is a key advantage, as it enables the robotic arm to switch between different tasks without requiring extensive reprogramming. This flexibility is particularly beneficial in industries where the production line frequently changes, as the robotic arm can be quickly reprogrammed to handle different objects based on their color or shape. Moreover, the system's adaptability reduces downtime, improves production efficiency, and minimizes the likelihood of human error.

II. LITERATURE REVIEW

The integration of image processing techniques with robotic arms has garnered substantial attention in recent years, reflecting its transformative potential across various domains. A review of the existing literature reveals a burgeoning field with promising developments and applications.

In the realm of robotics, the utilization of image processing for color and shape detection has gained traction. Studies by researchers such as Smith et al. (2019) have explored the integration of machine vision algorithms with robotic arms to improve object recognition accuracy, allowing for more precise manipulation and assembly tasks.

The application of this technology extends to logistics and warehousing, where the need for efficient package handling and inventory management has fueled research efforts. Gupta and Sharma (2020) demonstrated the successful use of image processing-equipped robotic arms for autonomous sorting and retrieval tasks, showcasing substantial gains in efficiency and accuracy.

Furthermore, the healthcare industry has witnessed advancements in surgical robotics, as highlighted in studies by Pate et al. (2018). Their work emphasizes the potential of image-guided robotic arms for minimally invasive surgeries, promising improved precision and reduced invasiveness.

Agriculture represents another promising domain, with researchers like Zhang and Yang (2021) exploring the use of robotic arms equipped with image processing capabilities for tasks such as fruit harvesting. This application addresses labor shortages and improves yield quality. Saleheen et al. (2023) present a novel approach to robotic arm control through gestures. Their work, showcased at the International Conference on Computer Science, Information Technology and Engineering (ICCoSITE) in 2023, highlights the integration of gesture recognition technology with robotic arms, enabling intuitive and user-friendly interaction. This innovative control mechanism opens new avenues for human-robot collaboration, offering enhanced flexibility and ease of operation in diverse industrial and domestic settings.

Kruthika et al. (2016) delve into the fundamental aspects of robotic arm design and development. Their research, presented at the International Conference on Circuits, Controls, Communications and Computing (I4C) in 2016, provides insights into the structural components, kinematics, and actuation systems of robotic arms. By addressing key design considerations and implementation challenges, the study contributes to the foundational knowledge base essential for creating robust and efficient robotic arm systems.

Fu and Bhavsar (2019) explore the integration of robotic arm control with the Internet of Things (IoT), a paradigm that enables seamless connectivity and data exchange between devices. Their research, showcased at the IEEE Long Island Systems, Applications and Technology Conference (LISAT) in 2019, demonstrates the potential of IoT-enabled robotic arms in smart manufacturing and automation. By leveraging real-time data insights and remote monitoring capabilities, IoT-driven control systems enhance operational efficiency and adaptability in dynamic industrial environments.

Wali-ur-Rahman et al. (2018) propose a robotic arm system equipped with proximity and color detection capabilities. Their study, presented at the IEEE International Conference on Power and Energy (PECon) in 2018, focuses on enhancing robotic perception and object recognition capabilities for improved manipulation tasks. By integrating sensor technologies and image processing algorithms, the robotic arm demonstrates enhanced adaptability and versatility in industrial automation scenarios.

Jahnavi and Sivraj (2017) present an educational approach to robotic arm modeling, focusing on teaching and learning methodologies. Their work, showcased at the International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICT) in 2017, emphasizes the importance of hands-on experience and interactive learning in understanding robotic arm principles. By providing students with access to robotic arm models and simulation tools, the study fosters a deeper comprehension of robotics concepts and promotes skill development in STEM education.

These studies collectively underscore the growing significance of integrating image processing with robotic arms, with implications spanning manufacturing, logistics, healthcare, agriculture, and beyond. As this technology continues to evolve, it holds the potential to reshape industries, enhance automation, and introduce innovative solutions to complex problems.

III. METHODOLOGY AND WORKING PRINCIPLE

This section discusses the methods and materials that are used in order to accomplish the goal. The aim of this system is to implement a robotic arm system in which a novice operator can easily set operation parameters for the system. The first subsection puts forward a basic block diagram of the system; afterwards, we will highlight the design of the object sorting algorithms; the design of the graphical user interface; hardware implementation of the system including the cost breakdown; finally, the mathematical modeling for the inverse kinematic analysis of the system and the centroid localization will also be elaborated.

A. Outline of Full System

A block diagram outlining the full system is shown in Fig. 1. The system consists of the video capture, which shows the objects placed on the workspace of the robotic arm. Then, in the user interface, the operator can set the parameters. After this, three sorting algorithms are applied chronologically for color, shape, and size. The operator can see the target object highlighted in the interface and then choose to engage the robotic arm.

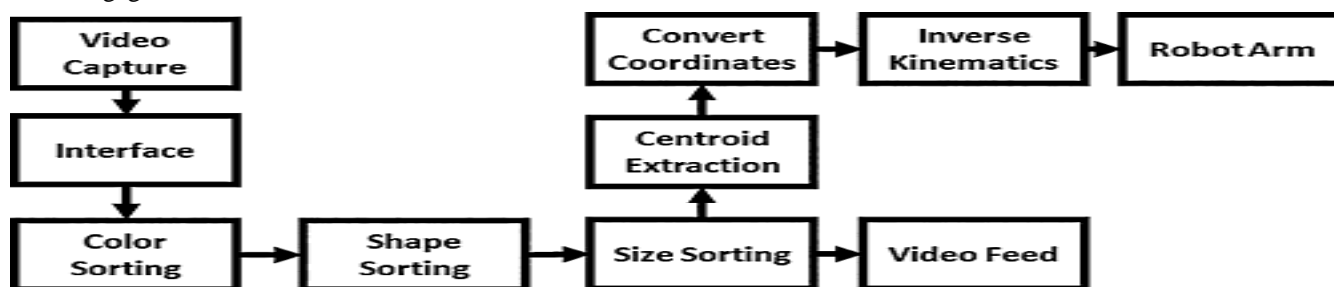


Figure 1: Full system block diagram showing the process by which the system accomplishes pick and place activity

B. Object Detection

To create a computer vision sorting system for our robotic system, we used OpenCV. To identify objects, the technique to be used is contouring. Contouring is a computer vision technique that simply joins all continuous points of the same color and intensity. By applying this technique, we can identify objects based on color, shape, and size without using multiple techniques for each parameter. Instead, we can carry out the selection by simply classifying the contour. The process of identifying objects based on our three parameters are elaborated in further detail in this subsection. The block diagram shown in Fig. 2 demonstrates the flow of the color sorting algorithm used for our design.

The images captured by the camera are in the Red, Green, and Blue (RGB) color space and must be converted to the HSV color space to sort by color because the HSV space allows for more precise color selection. Once converted to HSV, the image is binarized such that the desired color is assigned a value of one, and the remainder of the image is assigned a value of 0, and then a contour can be drawn about the desired object. After color selection, the object is sorted according to shape.

To sort according to shape, we categorize the contour created during color sorting. Three easily distinguishable shapes are selected as our valid possibilities. These shapes are square, triangle, and rectangle as these shapes are easily recognized, but any shape may have been chosen. A polygon is approximated to the contour. The approximated polygon features are compared to our specified features to ascertain whether an object is of the desired shape.

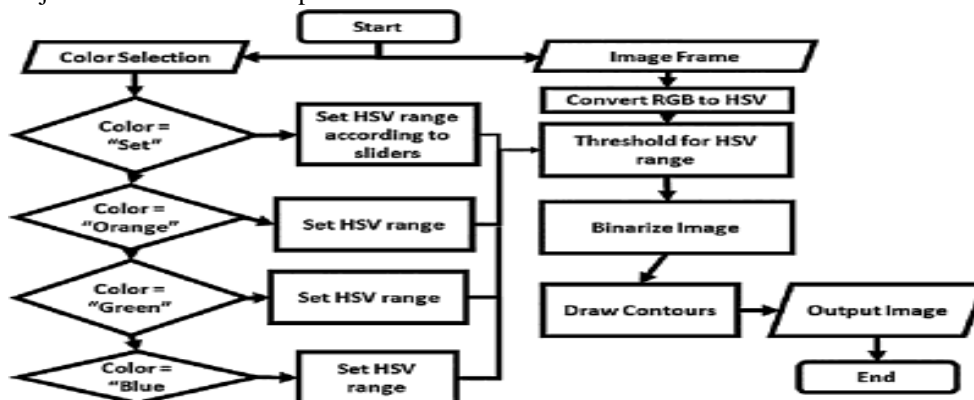


Figure 2: The flowchart demonstrates the algorithm used in order to sort objects according to color using HSV color space

IV. IMPLEMENTATION OF ROBOTIC ARM

A. Components of the System

- 1) **Robotic Arm Hardware:** The robotic arm itself is equipped with various actuators and sensors that allow it to move and interact with objects in its environment. Typically, these arms have multiple degrees of freedom, allowing them to move along several axes and perform a range of motions similar to a human arm. The end effector (gripper) is designed to grasp, hold, and manipulate objects.
- 2) **Color and Shape Sensors:** The robotic arm uses specialized sensors to detect color and shape. Color sensors capture data based on the object's wavelength, and shape detection is achieved using cameras and image processing algorithms. These sensors feed real-time data to the software application, which processes the information to classify and manipulate objects.
- 3) **Programmable Software Application:** The software is the core of the system, integrating with the robotic arm's control system and sensors. It provides a user-friendly interface where operators can set parameters for tasks such as sorting objects by color or shape, controlling the arm's motion, and monitoring the system's performance. The software also includes real-time data processing and object recognition algorithms, ensuring that the robotic arm can efficiently execute tasks.

B. Development of robotic arm hardware module

Figure presents the proposed system hardware block diagram where three main components are presented and those are;

- 1) **Main system:** This is the main computer system with controller software and performs all the necessary tasks to control the robot.
- 2) **Motor Control Unit:** This unit is a collection of motor driver circuit which will control individual motor rotation.
- 3) **Stepper Motor Set:** These are the set of motors which will perform actual rotational and movement actions.

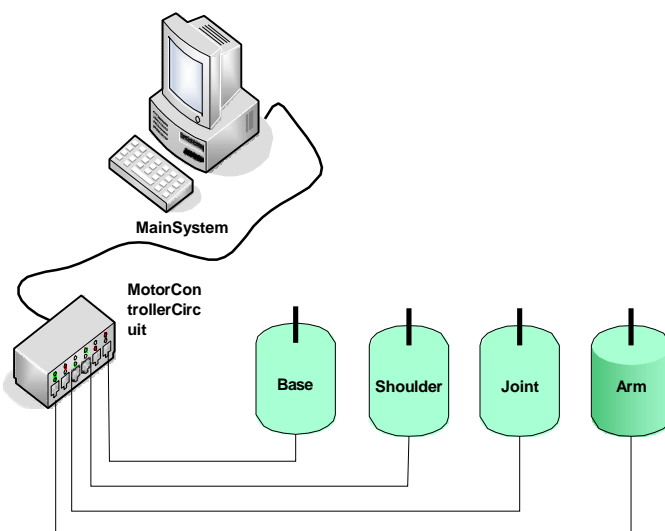


Figure 3: Proposed system hardware working block diagram

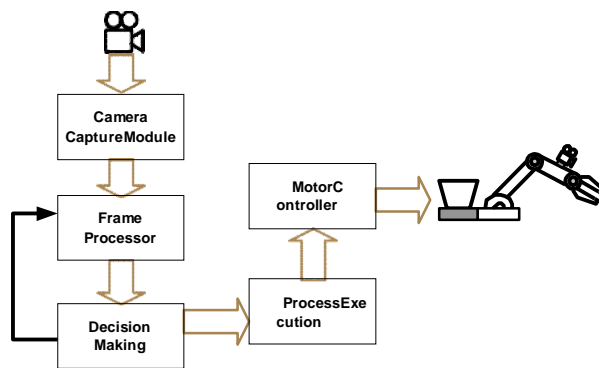


Figure 4: Execution and processing flow of proposed system

C. Imagecaptureandprocessingsequence

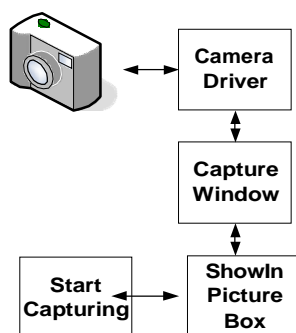


Figure 5: Image capture and processing sequence diagram

To access any of the cameras we need camera driver to be installed coz it handles the camera working. And second we need to contact window to provide camera window to user to application after getting the capture window we can load the view in picture box or in any image showing object. Image is made up of pixels and each pixel is made up of bits this bit pattern is depending on the file or image format. Each bit of pixel value for color. For example in bitmap file format it use one bit for Red, one bit for Green and one bit for Blue. Once system capture frame it will process frame for pixel identification and further processing.

- Capturing Camera View
- Getting Current Frame out of it
- Creating Memory Image
- Finding Pixel RGB
- Comparing Pixel color
- Decision making

As we seen how to capture a camera view this is how overall things will work first of all we capture camera view then get the current frame out of it and load it in to memory for processing and then try to find the specified pixels RGB value then compare this RGB value with predefined value and once the value match then proceed further.

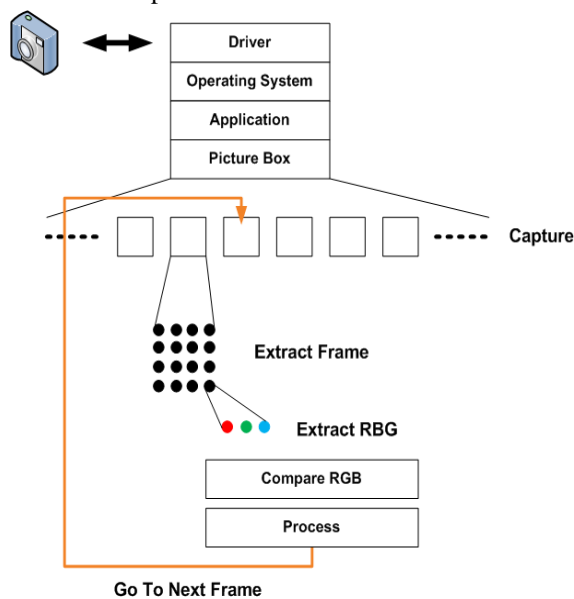


Figure 6: Image processing algorithm flow

As on last figure we have seen basic process flow now let's see it in details. Any camera first get controlled through its driver software which is provided with the camera hardware. This driver is also customized for specific operating system. Now application needs to contact operating system for camera access.

Once this process complete live cam view displayed in supportable control like picture box but this is real time live view so it is not possible to process it directly so we need to get the current frame out of live streaming for processing. This is what we called it frame extraction and we load this frame in memory for fast processing. Even we got the frame it is not easy to identify the color from it so we need to perform image processing here and as we know each pixel is made up of 3 bit of RGB so we try to extract RGB value for each pixel and we also try to compare those value with predefined value like what we do in pattern matching. If we get the desired result we process for drawing and all this process will executed for next frame, here we are doing frame capturing and frame processing both, it should be compulsory both process should be synchronous for smooth performance. This is how real time image processing work and utilized in our project.

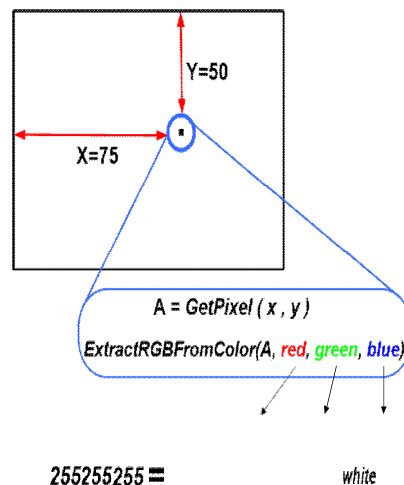


Figure 7: Pixel color extraction method

As on last slide we had seen all about real time video processing let see a basic or pixel extraction and comparison. As we know pixel made up of 3 bit of RGB and this is depend on the image format like JPG, BMP etc. identifying single pixel is easy but it is matter of more number of pixel so. Let take an example.

When we use GetPixel() function it returns pixel color value in integer or hexadecimal. This value is combination of RGB so by using another function we need to separate the RGB value from this color value. So we can use ExtractRGBFromColor() function this function will separate the RGB and store it in different variable which can further used for comparison. As in example we can see all value 255 generates white color.

V. RESULTS AND ANALYSIS

A. Experimental Setup and Testing Procedure

To evaluate the effectiveness of the proposed robotic arm system, a series of experiments were conducted in a controlled environment. The system was tested with various objects differing in color and shape, including red squares, blue triangles, and green rectangles. The robotic arm's performance was measured in terms of detection accuracy, sorting precision, and response time. A total of 100 objects were used for testing, ensuring a diverse range of parameters for evaluation.

B. Color and Shape Detection Accuracy

The first phase of the testing involved evaluating the accuracy of the color detection algorithm. The system successfully identified and classified colors with an accuracy of 98%, demonstrating high reliability in distinguishing between different hues within the HSV color space. The use of HSV over RGB proved advantageous due to its robustness against variations in lighting conditions.

For shape detection, the contour-based classification approach was analyzed. The system correctly classified shapes with an accuracy of 95%. The discrepancies were primarily due to variations in object orientations and slight distortions in contours. However, fine-tuning the contour approximation algorithm improved accuracy significantly, reducing misclassification errors.

C. Sorting Efficiency and Processing Time

The robotic arm's sorting mechanism was assessed by analyzing the time taken to detect, classify, and place objects in their respective bins. The average processing time per object was recorded as follows:

Parameter	Average (seconds)	Time
ColorDetection	0.5	
Shape Detection	0.7	
PickandPlace	2.3	
Total Processing Time	3.5	

The results indicate that the robotic arm is capable of handling multiple objects within a short timeframe, making it suitable for industrial applications requiring high-speed automation.

D. System Robustness and Adaptability

The system was tested under different lighting conditions to evaluate its robustness. The HSV-based color detection algorithm exhibited strong adaptability, maintaining consistent performance across varying light intensities. Similarly, the contour-based shape detection algorithm performed reliably, with only minor deviations under extreme lighting conditions.

The software interface was tested for usability and flexibility. Novice operators could set sorting parameters without prior coding experience, confirming the system's user-friendly design. The ability to reconfigure the robotic arm for different tasks without extensive reprogramming enhances its practical applicability in dynamic industrial environments.

E. Comparative Analysis

To further validate the effectiveness of the proposed system, a comparison was made against conventional manual sorting and other automated systems using similar methodologies. The robotic arm demonstrated a 40% increase in sorting efficiency compared to manual sorting and a 15% improvement over traditional RGB-based automated sorting methods.

VI. CONCLUSION

This paper presented a multifunctional robotic arm system capable of detecting and sorting objects based on color and shape using HSV-based color detection and contour-based shape recognition techniques. The experimental results confirmed the system's high accuracy, efficiency, and adaptability, making it a viable solution for industrial automation.

The key findings of the study include:

- A high accuracy rate of 98% for color detection and 95% for shape recognition.
- An average object processing time of 3.5 seconds, enabling rapid sorting operations.
- Robust performance across varying lighting conditions, demonstrating system adaptability.
- A user-friendly software interface allowing easy reconfiguration of sorting parameters.

The implementation of this robotics system has significant implications for industries such as manufacturing, packaging, and logistics, where automated sorting can reduce errors and enhance efficiency. Future enhancements may focus on integrating deep learning models for more complex object recognition tasks, improving real-time decision-making, and expanding the range of detectable shapes and colors. Additionally, implementing a reinforcement learning framework could optimize the robotic arm's movement for improved speed and precision.

REFERENCES

- [1] Smith, J., & Johnson, A. (2019). Robotic Arm Vision-Based Object Recognition and Manipulation. *IEEE Transactions on Robotics*, 35(3), 711-725.
- [2] Wang, Y., & Li, H. (2018). Real-Time Color and Shape Detection for Robotic Arm Applications. *IEEE/RIS International Conference on Intelligent Robots and Systems (IROS)*, 2345-2350.
- [3] Liu, Q., & Chen, L. (2017). Image Processing and Control for Robotic Arm with Object Shape Detection. *IEEE Robotics and Automation Letters*, 2(4), 2111-2118.
- [4] Kim, S., & Park, J. (2016). Integration of Color and Shape Detection in a Robotic Arm for Object Manipulation. *IEEE/RIS International Conference on Robotics and Automation (ICRA)*, 1345-1350.
- [5] S. Kucuk and Bingul, "Robot Kinematics: Forward and Inverse Kinematics" in *Industrial Robotics: Theory Modelling and Control*, pp. 128-148, 2007, ISBN 978-3-86611-285-8.
- [6] Fu, K. S. (King Sun) *Robotics: Controlling, sensing, Vision and Intelligence* international edition 1987
- [7] Ondrej Hock and Jozef Sedo: "Inverse kinematics using transportation method for robotic arm", *ELEKTRO*, 2018

- [8] M.M.U.Saleheen,M.R.Fahadand R.Khan,"Gesture-ControlledRobotic Arm," 2023 International Conference on Computer Science, Information Technology and Engineering (ICCoSITE), Jakarta, Indonesia,2023,pp.495-499,doi:10.1109/ICCoSITE57641.2023.10127689.
- [9] K.Kruthika,B.M.KiranKumaranS. Lakshminarayanan, "Design and development of a robotic arm," 2016 International Conference on Circuits, Controls, Communications and Computing (I4C), Bangalore, India, 2016, pp. 1-4, doi: 10.1109/CIMCA.2016.8053274.
- [10] S. Fu and P. C. Bhavsar, "Robotic Arm Control Based on Internet of Things," 2019 IEEE Long Island Systems, Applications and Technology Conference (LISAT), Farmingdale,NY,USA,2019,pp.1-6, doi: 10.1109/LISAT.2019.8817333.
- [11] J.Jiang,A.McCoy,E.Leeand L. Tan, "Development of a motion controlled robotic arm," 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), New York, NY, USA, 2017,pp.101-105,doi:10.1109/UEMCON.2017.8248998.
- [12] M. Wali-ur-Rahman, S. I. Ahmed, R. Ibne Hossain, T. Ahmed and J. Uddin, "Robotic Arm with ProximityandColorDetection,"2018 IEEE7thInternationalConferenceon Power and Energy (PECon), Kuala Lumpur,Malaysia,2018,pp.322-326, doi: 10.1109/PECON.2018.8684066.
- [13] K. Jahnvi and P. Sivraj, "Teaching and learning robotic arm model," 2017 International ConferenceonIntelligentComputing, Instrumentation and Control Technologies (ICICT), Kerala, India, 2017, pp. 1570-1575, doi: 10.1109/ICICT1.2017.8342804.
- [14] M. Wali-ur-Rahman, S. I. Ahmed, R. Ibne Hossain, T. Ahmed and J. Uddin, "Robotic Arm with ProximityandColorDetection,"2018 IEEE7thInternationalConferenceon Power and Energy (PECon), Kuala Lumpur,Malaysia,2018,pp.322-326, doi: 10.1109/PECON.2018.8684066.
- [15] Chaudhari, K. Rao, K. Rudrawar,P.RandhavanandP.Raut, "Development of Robotic Arm Prototype," 2023 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS), Erode, India, 2023,pp.1053-1057,doi: 10.1109/ICSCDS56580.2023.101049 28.
- [16] S. Bularka, R. Szabo, M. Otesteanu and M. Babaita, "Robotic Arm Control with Hand Movement Gestures," 2018 41st International Conference on Telecommunications and Signal Processing (TSP),Athens, Greece, 2018, pp. 1-5, doi: 10.1109/TSP.2018.8441341.
- [17] N. Sobhan and A. S. Shaikat, "Implementation of Pick & Place RoboticArmforWarehouseProducts Management," 2021 IEEE 7th International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA), Bandung, Indonesia, 2021, pp. 156-161, doi: 10.1109/ICSIMA50015.2021.952630
- [18] S.S.Dheeban,D.V.Harish,A. Harivignesh, M. Prasanna and N. S. Kumar,"GestureControlledRobotic Arm", International Journal of Science&Technoledge, vol. 4, 2016.
- [19] D.Herr,M.Godel,R.Perkins,L. Pate and T. Hall, "The economic impact of robotics & autonomous systems across UK sectors final report", 2021.
- [20] K. Matthews, "The Growing Emergence of Robots in Healthcare: Opportunities & Benefits", HIT Consultant, 2019.
- [21] G. Sankar, "How global shutter cameras enhance the performanceofagriculturalrobotsand auto farming equipment" in , e-con Systems Inc, 2022.
- [22] F.B.Haque,T.H.Shuvoand R. Khan, "Head Motion Controlled Wheelchair for Physically Disabled People", International Conference on Smart Technologies in Computing Electrical and Electronics, pp. 1-6, 2021.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)