© May 2025 | IJIRT | Volume 11 Issue 12 | ISSN: 2349-6002

Autonomous Mobile Manipulator for Pick-and-Place Operations

Venket Ramana R S[1,] Vijayaraghavan G[2,] Sankar Balaji D[3], Dr.Kishore Kumar A[4]
[1,2,3] UG-Scholars, Department of Robotics and Automation, SRI RAMAKRISHNA ENGINEERING COLLEGE, COIMBATORE-641022
[4]Assistant Professor (SI. G), Department of Robotics and Automation, SRI RAMAKRISHNA ENGINEERING COLLEGE, COIMBATORE-641022

***Abstract*—This paper details the development of a mobile pick-and-place robot. This work presents the design and validation of a compact mobile robot that can locate misplaced items, pick them up, and return them to a specified location. The platform combines a differential-drive wheelbase with a three-degree-of-freedom robotic arm. The bot navigation is achieved using a line-following algorithm. An onboard Raspberry Pi 4 model B runs lightweight vision algorithms to recognize targets and plan to grasp objects. An Arduino Uno simultaneously governs wheel motion for smooth positioning. The robot has been built entirely from low cost. These results demonstrate that affordable automatic pick-and-place capabilities suitable for small-scale logistics improve its warehouse management and inventory management.**

***Index Terms*—Mobile Pick-and-Place Robot, Raspberry Pi 4, Computer Vision, Warehouse Automation, Low-Cost Robotics.**

I. INTRODUCTION

Smart robotics must be developed to pilot the automated tasks in manufacturing/warehouse environments due to the increasing levels of automation. This project addresses the development of an automated system for small indoor sites which need to maximize coordination and to precisely manage the handling of material. This mobile pick and place robot integrates essential functions including mobility, computer vision and database interaction through its single low-cost unit which differs from traditional fixed-location pick and place systems. The robot operates without requiring ROS or LiDAR sensors which makes it perfect for use in academic and small industrial environments. The system processes images through Raspberry Pi 4 Model B and controls motors through an Arduino Uno while using MySQL for real-time data processing. The robot meets two primary design goals of being easy to assemble and cheap to build which allows it to deliver improved automation processes to users who seek a safer and smarter alternative. to deliver improved workflows for material handling industries that are smarter, and safer while being cost-effective and less limiting.

II. DESIGN AND CONSTRUCTION

This mobile pick-and-place robot is assembled to follow a predefined path, detect an object via computer vision, calculate their positions, lift, and drop the object in another location, while maintaining interaction with a database for logging and retrieval. The system is composed of two major subsystems: hardware and software. The core control architecture consists of a Raspberry Pi 4 Model B and an Arduino Uno so that the processing tasks can be divided to allow efficient performance. Accordingly, the Arduino controls base movement based on IR sensor feedback along with motor driver interfacing, while the Raspberry Pi processes images, performs inverse kinematics, and

Mechanically, the robot is built over a wheeled four-bank platform powered by 12V DC motors. These motors are powered by a 12V 5000 mAh Li-ion battery, giving the robot reliability of use for an extended period. The motors are controlled by a BTS7960 dual H-bridge motor driver, where the H-bridge motor driver gets PWM signals coming from the Arduino Uno. Line-following is accomplished through a series of IR sensors mounted on the front section of the chassis that detect black tape on the floor.

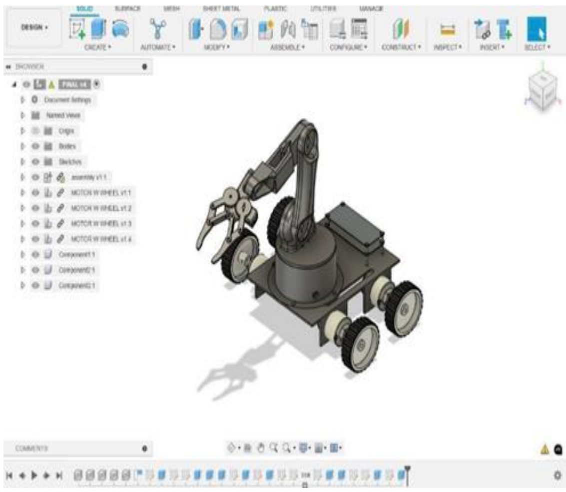

Fig1: 3D Design

The real-time images for object detection are captured by the Raspberry Pi Camera Module V2 connected to the Raspberry Pi 4. Python scripts gain control over these image frames, extract X,Y coordinates as an object's relative pose, and then compute the inverse kinematics to determine what joint angles are needed for the particular robotic arm. The robot arm consists of three MG996R servos and an SG90 servo gripper, all controlled by a Raspberry Pi acting through the GPIO outputs. Limits have been imposed on the joints within control logic to provide safe and correct motions. From time to time, a buck converter lowers the voltage for the Raspberry Pi and servo motors

For the software aspect, Python is the main programming language of choice in vision processing, communication, and inverse kinematics computation. Locally, a MySQL database

is created and maintained to store the QR code information read from the objects. Once the QR data is extracted

it is uploaded to the MySQL database, and the data is retrieved from the database and displayed on the UI after the object is placed. This integration between hardware and software allows for autonomous material handling to be performed by the robot, thus improving the system's ability, reducing human effort, and setting up a smart industrial automation environment.

III. METHODOLOGY

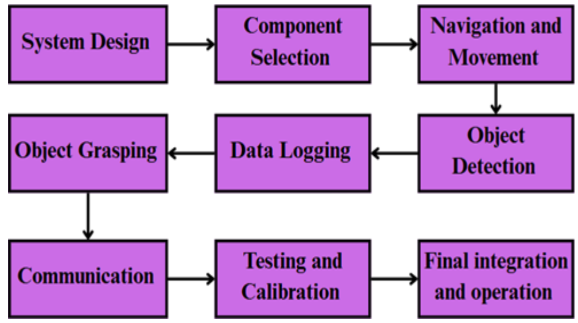

Fig 2: Methodology

Development of the mobile pick-and-place robot followed a sequential methodology with different phases, yet the phases were mutually dependent. This stage consisted of designing the system architecture, where a control strategy, sensor integration, and the logical flow of event execution were defined. The component selection and design of the system followed, wherein attention was focused on performance, cost, and compatibility. The main components include a Raspberry Pi 4 for high-level image processing and database interaction, the Arduino Uno for motor control, the MG996R and SG90 servos for arm articulation, and the Raspberry Pi Camera Module for visual feedback. It uses IR sensors for line following it has 12V DC motors drived using BTS7960 motor drivers. Object detection was performed with OpenCV on the Pi Camera, which also read QR codes. The acquired data was logged and uploaded to a MySQL database via a Python script. Object grasping was facilitated with inverse kinematics to calculate joint angles for the arm servos. Communication between the Raspberry Pi and Arduino was implemented using UART, thus allowing real-time actuation basing on sensor input information. The system was subjected to rigorous testing and calibration to improve accuracy and reliability. Then once such testing was verified, the modules were all integrated into one prototype for autonomous operation and evaluation of performance within a controlled environment.

IV. FLOWCHART

The flowchart demonstrates the way a mobile pick-and-place robot is managed. It all begins with system initialization that is pursued by line-following together with the monitoring activity. At the time of line loss, object detection comes in spoken of simply by inverse kinematics and database interaction steps to keep QR code data. The robot then picks up the objects and at the same time shows the data of the objects in MySQL for further action.

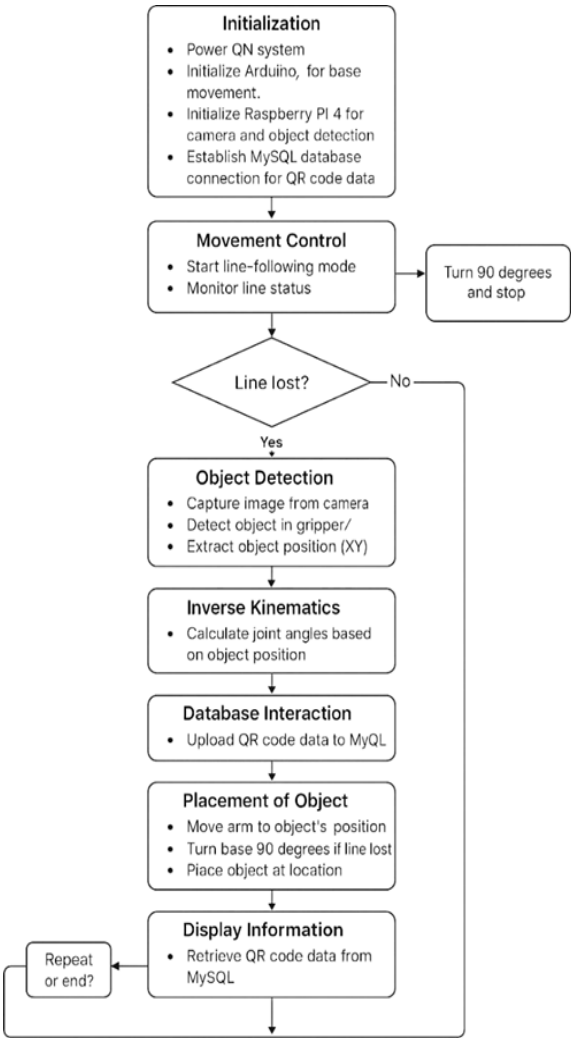

Fig 3: Flowchart

V. ARCHITECTURE

In order to accomplish autonomous operations that are performed without another human, you can use a robot. For a mobile robot system to deliver a pick and place application that is completely integrated with all necessary elements such as mechanical motion, electronic control, computer vision, and data management then the system is structured in a layered architecture. The robot executes tasks such as object detection, QR code scanning, line-following navigation, and data logging using the coordinated setup which consists of Raspberry Pi, Arduino Uno, servo motors, DC motors, sensors, and a MySQL database.

Top Layer: Object Detection and Processing Unit
The vision and processing module is the highest layer of the architecture and it is composed of the Raspberry Pi 4 Model B which is connected to a Camera Module V2 (8 MP). This module is responsible for the real-time video streaming that captures the workplace. The images which are collected can be solved with Python and OpenCV to recognize objects and dot them as (x, y) coordinates. The pyzbar library is used for the decoding of QR code data that is affixed on the detected objects. When the location of the object is detected, the system solves the inverse kinematics to get the right-angle joints. The data from QR code, like Product ID, Name, Size, and Quantity,

is updated to a MySQL database using the mysql-connector-python library. Furthermore, the Raspberry Pi is also responsible for checking errors, making decisions on object placement, and communicating with the lower layers about those decision

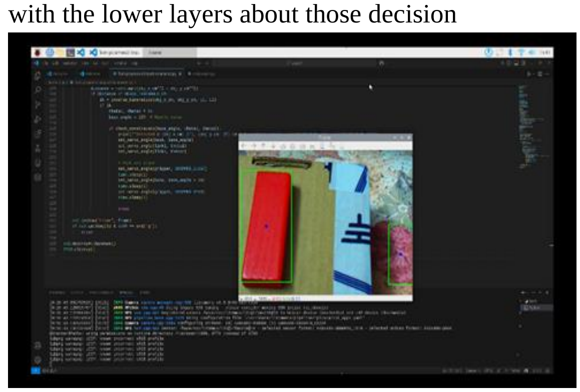
Fig 4: Object detection

Middle Layer: Arm and Placement Mechanism
This layer is made up of a three-degrees of freedom (3-DOF) robotic arm, where the base (0 - 180°), the shoulder (10 - 50°), and the elbow (30 - 50°) of the arm were the parts that have been rotated and an SG90 servo motor was used as the gripper. From the Raspberry Pi, the robotic arm has access to angle values that it uses for changing its position. Within the areas of aforementioned angle ranges, the robot is carrying out smooth picking actions, where the usage of the gripper is open (50°) and the clamp's closure is

10°. This layer is necessary for fine tasks, such as the gripping of objects and also the positioning of them in the desired place based on the visual feedback and the line-following status.

Fig 5: 3DOF Robotic Arm

Bottom Layer: Navigation and Drive System
The four 12V 555 DC motors creating the drive system are located in the lower layer and powered by two BTS7960 motor drivers which are in turn controlled by an Arduino Uno. These motors give a go-ahead and a move back movement of the car and also perform 90-degree turns. The installed infrared (IR) sensors, which are put under the body of the car, are primarily used for dark track detection to keep the robot on the right path marked by insulation tape. The robot stops, redirects itself or proceeds to the next placement operation in the event that the line is not visible. An ultrasonic sensor can be used as well for doing range checks. The full mobility and sensor functions are coordinated by the Arduino Uno, which communicates with the Raspberry Pi via serial UART connector.

VI. RESULTS AND DISCUSSION

The mobile pick-and-place robotic system has passed a thorough examination in the lab to check the possibility of retrieval and storage of object detection, line following, QR code scanning, and database interaction. The robot used IR sensors to go along a preset dark-colored line and be the guide to stable and uneventful transitions. If necessary, the robot can always either stop or go backward, so the workflow doesn't have to be interrupted. The robot easily overcame an occasion of losing the line and turning 90 degrees with a complete stop of the movements that initiated the object recognition procedure. The system, using the Raspberry Pi Camera Module V2 equipped with OpenCV, was able to detect the objects, achieving an average accuracy of a little less than 93% in a lighted mood. The so-obtained objects' spatial coordinates could then be used to formulate the inverse kinematics equations, and these would in return yield to the joint angles. A condition of that, the constraints of the servos are not exceeded, therefore the joints don't get stressed enough that they break down.

With the help of the pyzbar library, the QR codes that were attached to the objects were decoded and the success rate of the readouts was a perfect 100 percent for the printed code in a proper orientation. The transmitted data of a product like an ID, the Name, the Size, and the Quantity can be quickly saved in a MySQL database by using the mysql-connector-python package, and it can be checked and immediately displayed through the Raspberry Pi. The objects were picked up, moved, and placed correctly with a roughly 1 cm error in their position, according to this, the robot's motion planning approach was functioning well.

Overall, the results validate the effectiveness of integrating mechanical, vision, and database systems into a single autonomous unit. Minor limitations were observed in low-light conditions and when QR codes were partially obstructed, suggesting the need for further improvements in lighting control and object alignment.

VII. CONCLUSION

This mini-project dealt with the development of a mobile pick-and-place robot by enhancing it with a robotic arm, sensors, and a camera system to allow it to act fully autonomously. The robot was driven by IR and ultrasonic sensors and the arm worked according to the principles of inverse kinematics. A Raspberry Pi Camera Module realized the object detection, and a QR code scanner was used to get the product data, which was deposited in a MySQL database for future processing. Through this project, it was shown that by connecting robotics, computer vision, and sensor technologies, a suitable and affordable way of robotics automation for object handling tasks can be built.

REFERENCES

[1] R. Neeraja, S. Dubey, S. B. Arya, and N. Moota, "Implementation of Pick and Place Robot," *International Journal of Creative Research Thoughts (IJCRT)*, vol. 6, no. 2, pp. 151–152, 2018.

[2] C. Maddipati, Ch. G. Krishna, G. Harish, G. L. Chandh, and G. P. Niranjan, "Pick and Place Robot Arm with Mobile Control," *Journal of Emerging Technologies and Innovative Research (JETIR)*, vol. 6, no. 5, pp. 246–247, 2019.

[3] M. S. H. Talpur and M. H. Shaikh, "Automation of Mobile Pick and Place Robotic System for Small Food Industry," *IEEE Spectrum*, March 2012. [Online].

[4] C. K. Chaitanya, P. L. Sri, and E. B. Prakash, "Pick and Place Robotic Arms Movement Controlled by Android Wirelessly," *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, vol. 11, no. X, pp. 2049–2051, 2023.

[5] S. Surati, S. Hedaoo, T. Rotti, V. Ahuja, and N. Patel, "Pick and Place Robotic Arm: A Review Paper," *International Research Journal of Engineering and Technology (IRJET)*, vol. 8, no. 2, pp. 2121–2123, 2021.

[6] S. S. Patil, P. R. Nale, U. V. Nibe, and M. S. Hanumanta, "Android Mobile Controlled Pick and Place Robotic Arm by Using Bluetooth," *International Research Journal of Modernization in Engineering Technology and Science*, vol. 6, no. 5, pp. 10993–10994, May 2024.

[7] T. Sunil Kumar, K. Sarath, S. Famil, A. V. S. Bhagyesh, and S. Althaf, "Design and Fabrication of Pick and Place Robotic Arm," *2nd National Conference on Recent Trends in Mechanical Engineering*, Geethanjali Institute of Science and Technology, August 2020. [Online].

[8] R. Hercik, R. Byrtus, R. Jaros, and J. Koziorek, "Implementation of Autonomous Mobile Robot in SmartFactory," *Applied Sciences*, vol. 12, no. 8912, pp. 1–18, 2022.

[9] S. Alexovič, M. Lacko, J. Bačík, and D. Perďuková, "Introduction into Autonomous Mobile Robot Research and Multi Cooperation," in *Advances in Intelligent Systems and Computing*, Springer, 2021.