MAIT Journal of Science & Technology
Issue No. 1, Volume No. 2, December 2025, Page No.35-47
ISSN(Online)

Vision-Guided Autonomous Holonomic Robot for Dynamic Indoor Environments

Parth Sharma*, Neelu Nagpal, Neha Aggarwal
Electrical and Electronics Engineering Dept., Maharaja Agrasen Institute of Technology, Delhi
* Corresponding Author. Mob.: (+91) 7428560844, E-mail: parthsharma3101@gmail.com

Abstract:*This work presents the design and development of an autonomous holonomic drive robot system with object detection capability, enabling movement in all directions using computer vision and Mecanum wheels. Computer vision-based object detection allows the robot to recognize its surroundings, categorize objects, and respond promptly, while the drive system facilitates smooth movement of the robot across its entire workspace. The capabilities of the robot are further enhanced using OpenCV and YOLO technology to enable effective movement and interaction in changing environments. A two-tier control architecture integrating a Raspberry Pi Zero 2W and an ESP8266 microcontroller effectively isolates sensitive vision hardware from high-current motor components through a dedicated driver circuit. Trajectory tracking and system validation are carried out through Gazebo simulation. The result is a successfully developed autonomous robot prototype capable of navigating dynamic indoor environments without reliance on GPS infrastructure, making it well suited for a range of real-world applications.*

Keywords: Autonomous Robots, Holonomic Drive, Computer Vision, OpenCV, YOLO.

1. INTRODUCTION

Robots are an integral part of automation that can work independently in complicated environments with ease. The main aim of robot is object detection to recognize barriers, allowing the robot to move around and steer clear of possible obstacles successfully. Holonomic drive systems integrated with computer vision overcome these restrictions by enabling robots to navigate smoothly in all directions and make intelligent choices based on their environment. These features are useful to deploy this robot to various fields, such as warehouse automation and aiding in rescue missions. This research is centered on creating a self-navigating robot with omnidirectional drive for identifying objects. The next section presents the literature survey, gaps identified, and main contributions of proposed work.

2. LITERATURE REVIEW

A literature survey has been conducted to check progress in autonomous robotics. There has been a notable advancement in holonomic drive systems, omnidirectional mobility. It has been reported that computer vision provides better navigation and perception. The review summary in Table 1 combines main findings from important works. From the reported work, the gaps are identified providing the motivation behind this present research.

Table 1: Summary of Reported Work

Ref. No. & Publication / Source	Title	Key Contribution
Doroftei et al. (2007) [1]	Omnidirectional Mobile Robot – Design and Implementation	Established foundational principles for the mechanical design and control of omnidirectional mobile robots; demonstrated that holonomic systems allow free directional movement without reorientation.
Mohd Salih et al. (2006) [2]	Designing Omni-Directional Mobile Robot with Mecanum Wheel	Explored the engineering challenges involved in building a Mecanum-wheeled mobile robot; highlighted the wheel's unique capacity for simultaneous lateral and forward motion.
De Silva & Munasinghe (2009) [3]	Development of a Holonomic Mobile Robot for Field Applications	Addressed the creation of a holonomic robot suited to outdoor/field conditions; emphasized the significance of terrain adaptability for practical autonomous deployment.

Ref. No. & Publication / Source	Title	Key Contribution
Tatar et al. (2014) [4]	Design and Development of an Autonomous Omni-Directional Mobile Robot with Mecanum Wheels	Detailed the hardware–software integration of an autonomous omni-directional platform; contributed practical insights into sensor fusion and control algorithm design.
Piemngam et al. (2019) [5]	Development of Autonomous Mobile Robot Platform with Mecanum Wheels	Examined design considerations for a Mecanum-wheeled autonomous robot and demonstrated how computer vision enhances obstacle identification and avoidance capability.
Taheri et al. (2015) [6]	Kinematic Model of a Four Mecanum Wheeled Mobile Robot	Formulated the mathematical kinematic model relating individual wheel angular velocities to the robot's linear and rotational motion, providing a rigorous theoretical basis for Mecanum-based control.
Kowalczyk & Joon (2022) [8]	Design of One Degree of Freedom Rotating Platform for Leader-Follower Autonomous Robot Control	Introduced a rotating platform mechanism for coordinating multi-robot leader–follower behaviour; underlined the importance of inter-robot communication in distributed systems.
Tsiakas et al. (2023) [9]	An Autonomous Navigation Framework for Holonomic Mobile Robots in Confined Agricultural Environments	Proposed an adaptive navigation framework for holonomic robots operating in constrained agricultural spaces; highlighted challenges specific to domain-specific autonomous deployment.
Toribio et al. (2021) [10]	Mobile Robot Localization in Industrial Environments Using a Ring of Cameras and ArUco Markers	Demonstrated the use of ArUco marker arrays with a multi-camera setup for precise industrial robot localization; contributed robust positioning methods applicable to automation workflows.
Mutalib & Azlan (2020) [11]	Prototype Development of Mecanum Wheels Mobile Robot: A Review	Reviewed iterative prototype development practices for Mecanum-wheeled robots; offered practical guidance on testing methodologies during the engineering lifecycle.
Asus et al. (2020) [12]	Design and Control of a Motor Wheel System Controlled by Smartphone	Described a smartphone-controlled motor-wheel system and presented a torque/force calculation methodology that serves as a reference for drive system parameter design.
Kumar et al. (2020) [13]	Autonomous Navigation of Holonomic Mobile Robot Using Point-to-Line Algorithm	Introduced a point-to-line path-planning algorithm for holonomic robots; illustrated improvements in manoeuvrability and collision-free navigation over conventional approaches.
Sheth et al. (2018) [14]	Design and Development of Intelligent AGV Using Computer Vision and Artificial Intelligence	Combined AI with computer vision to build an intelligent automated guided vehicle; emphasised the critical role of object detection in enabling reliable autonomous movement.
Garrido & Panov (2023) [15]	Detection of ArUco Markers – OpenCV Tutorials	Provided a step-by-step technical guide for implementing ArUco marker detection using the OpenCV library; a foundational reference for vision-based robot localisation.

The proposed work is driven by the increasing need for self-sufficient robots that can efficiently move around in changing environments and complete tasks independently and accurately without the need of GPS. The following are main research gaps:

- Most studies (Doroftei et al., de Silva & Munasinghe, Tatar et al.) have focused on outdoor or open-space navigation. Very few addressed reliable autonomous navigations in unstructured, dynamic indoor spaces without GPS.
- The work of Piemngam et al. (2019) and Sheth et al. (2018) have separately explored computer vision and Mecanum-wheeled robots, a complete integration of holonomic drive with real-time YOLO-based object detection on a single lightweight embedded platform is still largely unexplored.
- Most of the referenced systems (Puthussery et al., Kowalczyk & Joon) have relied on expensive depth cameras and high-end processors for navigation. A cost-effective alternative using a standard camera module with ultrasonic sensing for distance estimation is not well addressed.
- Existing works have typically used a single controller for both vision processing and motor control. The isolation of high-level vision tasks from low-level motor operations using a dual-microcontroller setup (Raspberry Pi + ESP8266) is not explored in prior literature.
- Kumar et al. (2020) and Tsiakas et al. (2023) have proposed navigation algorithms but few works have demonstrated Gazebo-based simulation validation before actual hardware testing, creating a gap in systematic development methodology.

2.1 Contributions:

The main work carried out is as follows:

- Development of full kinematic model derived for a 4-Mecanum-wheel robot linking wheel angular velocities to robot motion in X, Y, and rotational directions and appropriate selection of motors and battery based on torque computation.
- Development of a two-level electronics architecture isolating sensitive sensors from high-current motor drivers.
- Provision of ArUco marker-based indoor navigation as a GPS-free alternative, combined with SLAM concepts.
- Real-time object detection using OpenCV + YOLO for obstacle recognition and path guidance, and conduction of simulation study using Gazebo to validate ArUco navigation before hardware testing.

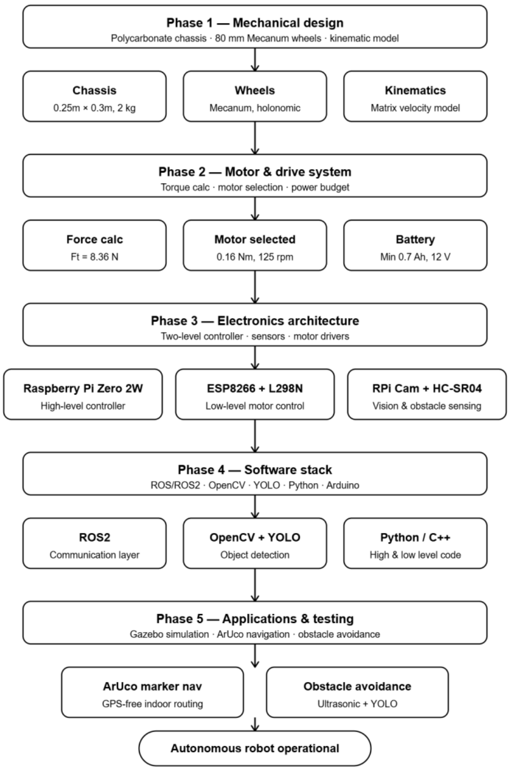


Fig. 1 Proposed Work Flow.

3. Proposed Work

The different phases of work is illustrated with the help of Fig. 1 and discussed in coming subsections.

3. System Design

3.1 Mechanical Design

(a) **Chassis design**: The chassis structure must be robust and is designed to accommodate Mecanum wheels, allowing for omnidirectional movement. To have these features we made our robot chassis from polycarbonate sheet because it is lightweight to support the necessary hardware without hindering mobility. The dimensions of the chassis are: Length: 0.25 meters, Width = 0.3 meters, Height = 0.15 cm, weight of the bot: 2kg

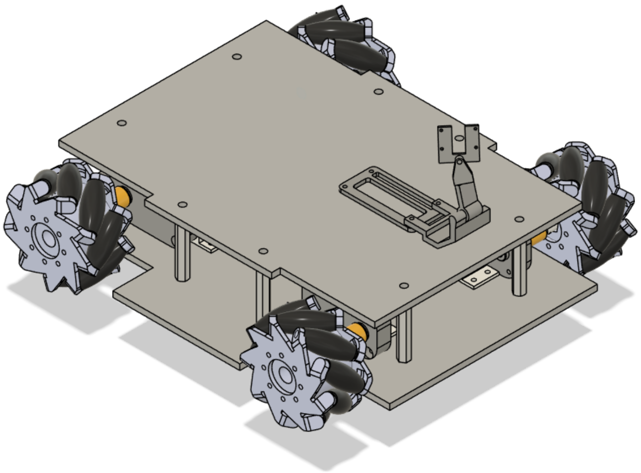
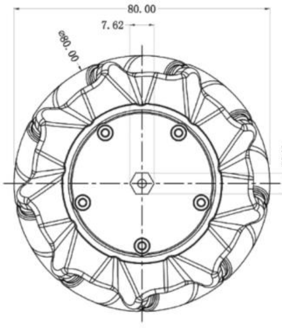


Fig 2 CAD Design of the Robot

a) **Wheel selection:** Mecanum wheels will be used to achieve holonomic motion, enabling the robot to move in any direction without changing its orientation. We have used 80mm Mecanum wheels:



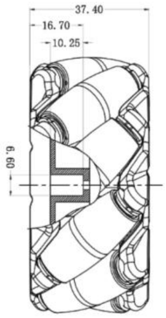


Fig 3 80 mm Mecanum rubber roller wheels

3.2 Kinematic modeling

The kinematic model of the robot is based on the velocities of the individual wheels and their contribution to the overall movement of the robot. The model uses a set of equations that relate the angular velocities of the wheels to the robot's linear and angular velocities in the X, Y, and rotational directions.

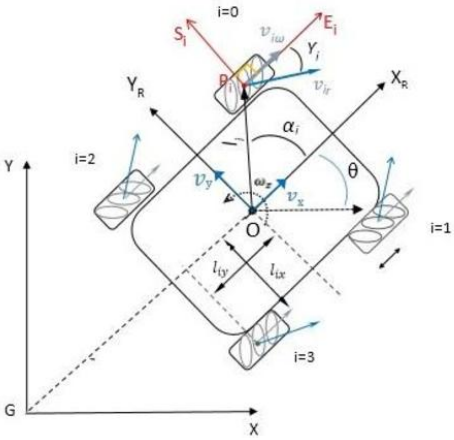


Fig 4 Wheels Configuration and Posture definition [6]

The robot's motion can be described by its velocity in the X (forward), Y (sideways), and θ (rotational) directions. The velocity of the robot $(\vec{V})$ is determined by the velocities of the individual Mecanum wheels (V_i), which are a function of their angular velocities (ω_i).

Each Mecanum wheel generates a velocity vector at a 45-degree angle to the wheel's plane. The robot's movement is the result of the superposition of these vectors. The kinematic model of the robot can be described using the following equations:

Let (V_x), (V_y), and (ω)represent the robot's velocity in the X (forward), Y (sideways), and rotational directions, respectively.

Let (ω_1), (ω_2), (ω_3), and (ω_4) be the angular velocities of the four wheels.

The relationship between the robot's global velocity vector and the individual wheel speeds is given by the following matrix equation:

$$\begin{bmatrix} V_x \\ V_y \\ \omega \end{bmatrix} = \frac{1}{4} \begin{bmatrix} 1 & -1 & -\frac{1}{L+R} \\ 1 & 1 & \frac{1}{L+R} \\ 1 & 1 & -\frac{1}{L+R} \\ 1 & -1 & \frac{1}{L+R} \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix}$$

Where, L is the distance between the front and back wheels, and R is the distance between the left and right wheels.

This matrix equation relates the angular velocities of the wheels to the robot's linear and angular velocities. By solving the inverse kinematics, we can determine the necessary wheel speeds to achieve a desired motion

3.3 Design of motor driver system parameters:

To calculate the torque required by the mobile robot to accelerate we first need to calculate the force required by the bot to accelerate. The primary criterion of designing a robot that carries load is to calculate the torque required to transmit the weight of the load from one point to another. Torque required for the system is calculated using the total tractive effort or force (F_t) of the system as in equation 1[12].

$$F_t = F_a + F_r + F_g + F_d \tag{1}$$

where, F_a = force due to acceleration, F_r = Force due to friction or wheel rolling resistance, F_g =force due to gradient resistance, F_d = aerodynamic drag. The wheel diameter is 80mm so the radius (r) will be 40 mm or 0.04 m. For one revolution of wheel, the robot will move a distance of:

$$2\pi r = 2 \times 3.14 \times 0.04 = 0.251m \tag{2}$$

The robot is designed to achieve a speed of 0.52 m/s.

In our experiment, we are assuming the robot is operating in an indoor environment and moving on a flat surface, so the F_g will be zero and aerodynamic drag is neglected due to an insignificant value. Therefore, the forces acting on the robot are F_a and F_r.

Force due to acceleration: $F_a = m \times a$ (3)

where, m = mass of the robot = 2kg; a = acceleration of the robot

assuming that the robot will achieve the desired velocity (V) of 0.52 m/s in time (t) 2 seconds, then the acceleration (a) of the robot can be calculated as:

$$a = V/t = 0.52/2 = 0.26m/s^2 \tag{4}$$

$$F_a = 2 * 0.26 = 0.52N$$

Force due to friction: $F_r = \mu \times N$ (5)

where, μ = coefficient of friction, N = Normal force

Assuming for coefficient of friction (μ) to be 0.4 for the floor and Normal force (N) acting on the robot is:

$$N = m * g \tag{6}$$

where m = mass of the robot = 2 kg and g = acceleration due to gravity = 9.8 m/s^2

So, $N = 2*9.8 = 19.6N$

Therefore, $F_r = 0.4*19.6 = 7.84N$

The total tractive force (F_t) required by the robot is:

$$F_t = F_a + F_r \tag{7}$$

$$F_t = 0.52 + 7.84$$

$$F_t = 8.36N$$

We need to select a permanent magnet DC motor that can provide the required force. The torque (τ) provided by the motor can be calculated using the formula:

$$\tau = F * r \tag{8}$$

where F is the total force required by the mobile robot = 8.36 N, and r is the radius of the wheel = 0.04 m.

$$\tau = 8.36*0.04 = 0.33Nm$$

Since our design is an omni-drive system with four mecanum wheels, each motor only needs to provide a quarter of the total torque. Therefore, torque per wheel or torque required by each motor is:

$$\tau = 0.33/4 = 0.08Nm \tag{9}$$

Then, with a design speed (V*)* of 0.52 m/s and wheel radius (*r)* of 0.04 m, we can calculate the required rated speed of the motor (*n*) in rpm as:

$$n = V*60/2\pi r \tag{10}$$

$$n = 0.52*60/0.251 = 124.3rpm$$

- To ensure that the robot can move reliably under all conditions, we would consider adopting a safety factor of 2, this means we will select a motor which can give a torque twice the calculated torque per wheel.

  Therefore, on the basis of above calculations and our robot design we need to select a permanent magnet DC gear motor with a rated torque τ = 0.16 Nm and with a rated speed of 125 rpm.

Fig 5 L-shaped permanent magnet dc gear motor

Based on the calculations above, assuming we select a permanent magnet DC motor with a rated voltage of E = 12V, rated torque of τ = 0.16 Nm, and rated speed of n =125 rpm, the required power of the motor can be calculated using the formula:

$$P = \tau \times \omega \tag{11}$$

where τ is the torque provided by the motor and ω is the angular velocity of the motor. Thus, Angular speed in rad/s (ω) of the motor and wheel will be:

$$\omega = n \times (2\pi/60) \tag{12}$$

$$\omega = 125 \times 0.104 = 13 rad/s$$

and, $P = \tau \times \omega = 0.16 \times 13 = 2.08W$

The above calculated required power is for one motor. The total power required for four motors is:

$$P_{total} = 4 \times 2.08 = 8.32W \tag{13}$$

and the rated voltage of the selected motors is E = 12V, we can calculate the required current (I):

$$I = P_{total}/E \tag{14}$$

$$I = 8.32/12 = 0.69A$$

Therefore, the minimum capacity of the power battery should be 0.7 Ah in order to power the robot for 1 hour of operation. If the robot needs to run for a longer period, a higher capacity battery should be selected.

From the above systematic calculation, we are able to select the right DC gear motor with 0.16 Nm, 125 rpm and battery capacity 0.7 Ah based on the torque computation.

3.4 Electronics Architecture

a) Microcontroller/Processor: A central processing unit, a Raspberry Pi zero 2W, will serve as the robot's brain, handling computations and decision-making.

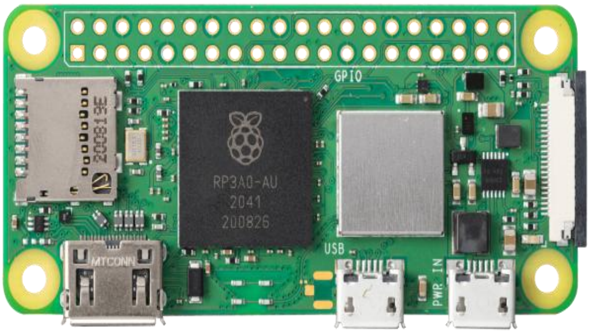


Fig 6 Raspberry Pi Zero 2W

b) Motor Driver Circuit: Motor drivers will be required to control the speed and direction of each Mecanum wheel independently. We have used two L298N Motor driver modules with ESP8266 for controlling the motors of our robot.

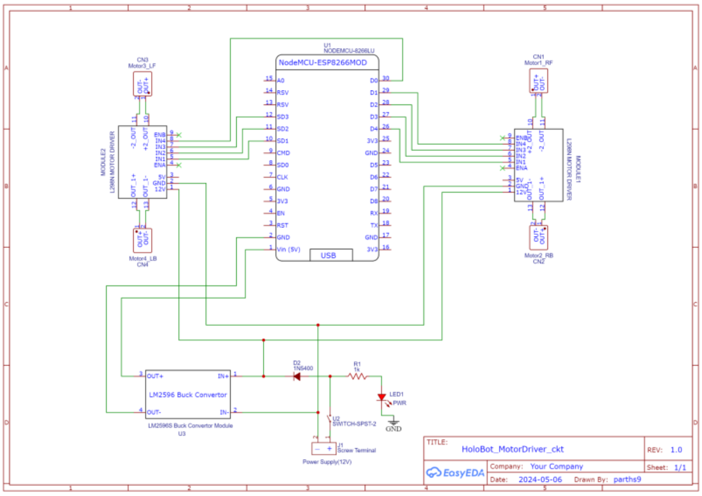


Fig 7 Motor Driver Circuit schematic

c) Sensors: A Camera module for object detection, we have used Raspberry Pi Cam for our project. We also have added an Ultrasonic sensor as an additional sensor for obstacle detection.

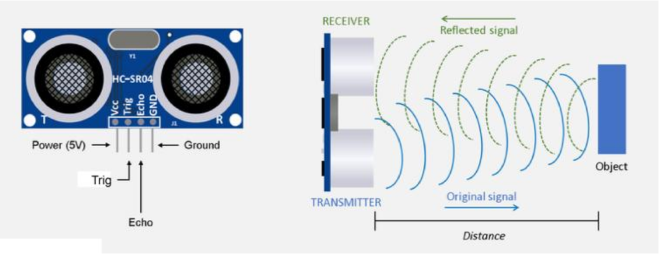


Fig 8 HC-SR04 Ultrasonic Sensor

Fig 9 RPi - CAM Module

Circuit:

The ultrasonic sensor and RPi - Cam module are connected to the Raspberry Pi Zero, which acts as a high-level controller, the Raspberry Pi can be connected to ESP8266 module either via WiFi or I2C interface. The motor driver circuit controlled by ESP8266 acts as a low-level controller. This two-level architecture helps in isolating sensitive sensor equipment from high-current components of the motor driver circuit.

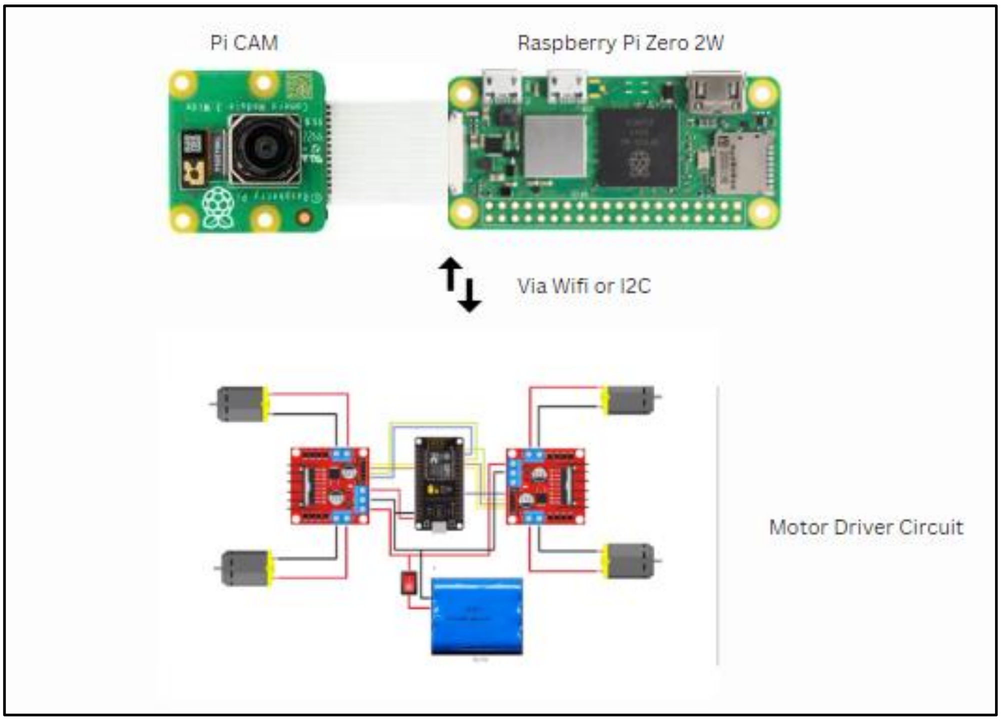


Fig 10 Electronics architecture of the robot

3.5 Software Stack

a) ROS/ROS2: The Robot Operating System (ROS) and ROS2 are middleware software packages for robotics that are open-source and offer services tailored for a diverse computer cluster. They provide resources and frameworks for acquiring, constructing, scripting, and executing code on various machines. ROS is commonly utilized in robotics research and development, offering an organized communication layer on top of the host OS in a diverse compute cluster.

b) OpenCV/YOLO: OpenCV is a wide-ranging open-source library used for computer vision tasks that provides a variety of functions for image processing and machine learning. YOLO, which stands for You Only Look Once, is a cutting-edge object detection system that operates in real-time and is built upon deep learning algorithms. As a pair, they are effective resources for assisting robots in accurately and quickly identifying objects and obstacles.

c) Python: Python is recognized for its simplicity and clearness, being a high-level, interpreted programming language. It is commonly selected for software development in different areas, like robotics, because of its wide range of libraries and frameworks like ROS and OpenCV, which make development and prototyping faster and easier.

d) C++/Arduino IDE Sketch: The ESP8266 microcontroller is programmed to control the motor driver circuit using Arduino IDE sketch, a subset of C++ programming language.

4. APPLICATIONS

4.1 ArUco Marker-Based Navigation

This combines aspects of beacon-based navigation and Simultaneous Localization and Mapping (SLAM), offering a balance between the two approaches.These stages are described below and illustrated with the help of Fig. 10.

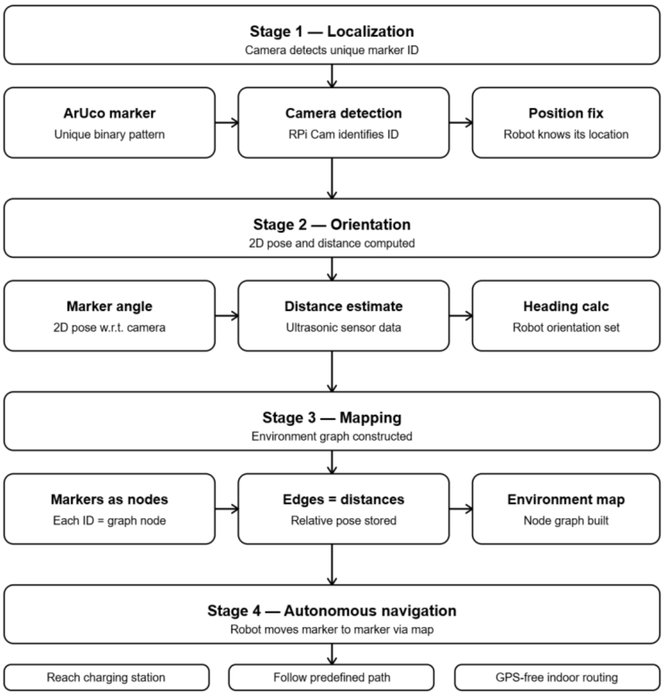


Fig 11 ArUco Marker-Based Navigation Stages

- **Stage 1** — Localization: camera detects unique marker ID
- **Stage 2** — Orientation: 2D angle + distance computed
- **Stage 3** — Mapping: node graph built from all detected markers
- **Stage 4** — Navigation: robot moves autonomously using the map

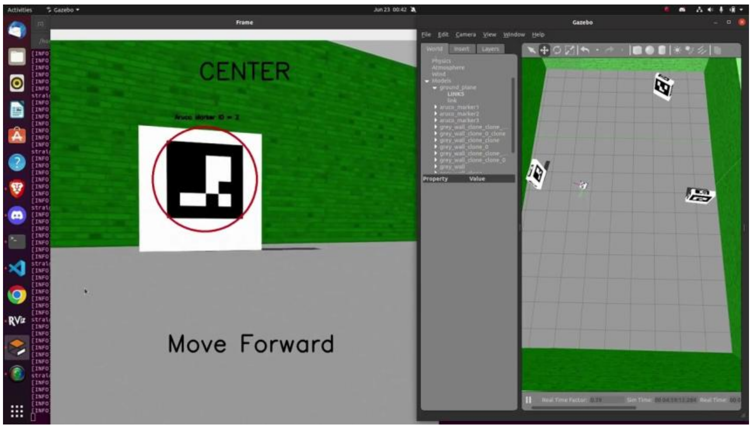


Fig 12 Simulation of Aruco Marker based Navigation in Gazebo Sim

In case. GPS signals are unavailable or unreliable, ArUco marker-based navigation is particularly used for indoor environments. From Fig 12, it can be seen that a depth camera is used to obtain the distance to the marker and relies on odometry calculations for tracking the robot's position after localization with respect to the marker. Since a depth camera was not available, the RPi Cam module was used for marker detection and an ultrasonic sensor for distance calculation. This allows the robot to navigate autonomously to various locations within the mapped environment, guided by the strategic placement of ArUco markers.

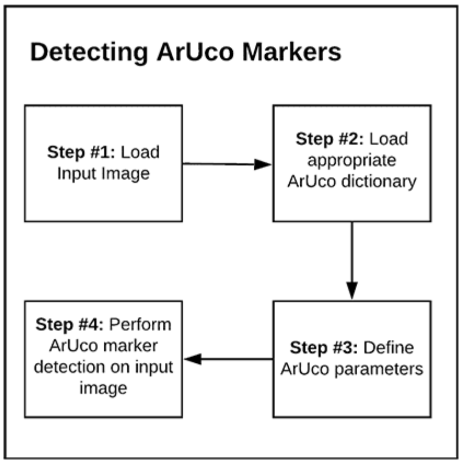


Fig 13 Flowchart of ArUco Marker Detection

5. CONCLUSIONS

This paper has successfully built a prototype of a self-sufficient holonomic drive robot capable to track a desired trajectory without the use of GPS. The approach of mechanical designing and OpenCV for software development was perfect, and the components integrated for performing various task to meet the objective of navigation was well-functioned. The computer vision system precisely detected and recognized objects. The functionality of each component was examined and the developed robot demonstrated precise navigation capabilities, especially in tight spaces. This has reflected the flexibility and potential influence of the proposed autonomous robots with advanced mobility and perception abilities. It is concluded that holonomic drive systems, computer vision, and adaptive navigation strategies functioned well for the progression of autonomous robot. In conclusion, this research provided a future direction to further explore and development of industrial robots. The future work will focus to develop holonomic drive robots with improved object detection algorithms and cost-effective solutions.

9. REFERENCES

[1] Doroftei, I., Grosu, V., & Spinu, V. (2007). Omnidirectional Mobile Robot - Design and Implementation. ref: https://api.semanticscholar.org/CorpusID:18429885

[2] Mohd Salih, J. E., Rizon, M., Bin Yaacob, S., Adom, A. H., & Mamat, M. R. (2006). Designing Omni-Directional Mobile Robot with Mecanum Wheel. American Journal of Applied Sciences, 3, 1831-1835. doi:10.3946/ajas.2006.3.1831

[3] de Silva, W. R., & Munasinghe, R. (2009). Development of a holonomic mobile robot for field applications. 2009 International Conference on Industrial and Information Systems (ICIIS), 499-504. doi:10.1109/ICIIS.2009.524605

[4] Tatar, M. O., Popovici, C., Mandru, D., Ardelean, I., & Plesa, A. (2014). Design and development of an autonomous omni-directional mobile robot with Mecanum wheels. 2014 IEEE International Conference on Automation, Quality and Testing, Robotics, 1-6. doi:10.1109/aqtr.2014.6857869

[5] Piemngam, K., Nilkhamhang, I., & Bunnun, P. (2019). Development of Autonomous Mobile Robot Platform with Mecanum Wheels. 2019 First International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP), 90-93. doi:10.1109/ICA-SYMP.2019.00028

[6] Taheri, H., Qiao, B., & Ghaeminezhad, N. (2015). Kinematic Model of a Four Mecanum Wheeled Mobile Robot. International Journal of Computer Applications, 113, 6-9. doi:10.1016/j.ijcai.2015.03.006

[7] Puthussery, A. R., Haradi, K. P., Erol, B. A., Benavidez, P., Rad, P., & Jamshidi, M. M. (2017). A deep vision landmark framework for robot navigation. 2017 12th System of Systems Engineering Conference (SoSE), 1-6.

doi:10.1109/SoSE.2017.804433

[8] Kowalczyk, W., & Joon, A. (2022). Design of one degree of freedom rotating platform applied in leader-follower autonomous robot control. 2022 26th International Conference on Methods and Models in Automation and Robotics (MMAR), 211-216. doi:10.1109/MMAR.2022.00028

[9] Tsiakas, K., Papadimitriou, A., Pechlivani, E. M., Giakoumis, D., Frangakis, N., Gasteratos, A., & Tzovaras, D. (2023). An Autonomous Navigation Framework for Holonomic Mobile Robots in Confined Agricultural Environments. Robotics, 12, 146. doi:10.3390/robotics12010146

[10] Toribio, S. R. H., Alvarez Garcia, I., & Gonz{'a}lez, R. C. (2021). Mobile robot localization in industrial environments using a ring of cameras and ArUco markers. IECON 2021 – 47th Annual Conference of the IEEE Industrial Electronics Society, 1-6. doi:10.1109/IECON.2021.942714

[11] Mutalib, M. A. A., & Azlan, N. Z. (2020). Prototype Development of Mecanum Wheels Mobile Robot: A review. ref: https://api.semanticscholar.org/CorpusID:229385782

[12] Asus, Z., Mat Razi, M. F. I., Mahmudin, M. F., Che Daud, Z. H., Mazali, I. I., & Ardani, M. I. (2020). Design and control of a motor wheel system controlled by smartphone. IOP Conference Series: Materials Science and Engineering, 884(1), 012097. doi:10.1088/1757-899X/884/1/012097

[13] Kumar, H., Zanzrukiya, R., & Adajania, V. K. (2020). Autonomous Navigation of Holonomic Mobile Robot Using Point-to-Line Algorithm. 2020 6th International Conference on Control, Automation and Robotics (ICCAR), 107-112. doi:10.1109/ICCAR50984.2020.920611

[14] Sheth, S., Ajmera, A., Sharma, A., Patel, S., & Kathrecha, C. (2018). Design and Development of Intelligent AGV Using Computer Vision and Artificial Intelligence. ref: https://api.semanticscholar.org/CorpusID:56608553

[15] Garrido, S., & Panov, A. (2023). Detection of ArUco Markers. OpenCV Tutorials, 4.9.0-dev. ref: https://docs.opencv.org/4.x/d5/dae/tutorial_aruco_detection.html