**[Autonomous Navigating and Interacting Robot]**

**Submitted**

**By**

**K.SAI GANESH REDDY – BU21EECE0100566**

**M.RAVIKIRAN REDDY - BU21EECE0100516**

**G.PRADEEP REDDY - BU21EECE0100084**

**Under the Guidance of**

**(M DIOLINE SARA** ASSISTANT PROFESSOR**)**

**(Duration: 20/08/2024 to 18/03/2025)**



**Department of Electrical, Electronics and Communication Engineering**

**GITAM School of Technology**

**GITAM**

**(DEEMED TO BE UNIVERSITY)**

**(Estd. u/s 3 of the UGC act 1956)**

**NH 207, Nagadenehalli, Doddaballapur taluk, Bengaluru-561203 Karnataka, INDIA.**

**DECLARATION**

**I/We declare that the project work contained in this report is original and it has been done by me under the guidance of my project guide.**

**Name:**

**(M DIOLINE SARA** ASSISTANT PROFESSOR**)**

**Date: Signature of the Student**

**18/03/2025**

K.SAI GANESH REDDY -BU21EECE0100566

M.RAVIKIRAN REDDY - BU21EECE0100516

G.PRADEEEP REDDY- BU21EECE0100084

**Department of Electrical, Electronics and Communication Engineering**

**GITAM School of Technology, Bengaluru-561203**

**A logo with text on it

AI-generated content may be incorrect.**

**CERTIFICATE**

**This is to certify that (K.SAI GANESH REDDY,M,RAVI KIRAN REDDY,G,PRADEEP REDDY) bearing (Regd.No.:BU21EECE0100566,BU21EECE0100516,BU21EECE0100084) has satisfactorily completed Mini Project Entitled in partial fulfillment of the requirements as prescribed by University for VIIIth semester, Bachelor of Technology in “Electrical, Electronics and Communication Engineering” and submitted this report during the academic year 2024-2025.**

**[Signature of the Guide] [Signature of HOD]**

**Table of contents**

[**Chapter 1: Introduction 1**](#_heading=h.gjdgxs)

[1.1 Overview of the problem statement 1](#_heading=h.30j0zll)

[1.2 Objectives and goals 1](#_heading=h.1fob9te)

[**Chapter 2 : Literature Review 2**](#_heading=h.3znysh7)

[**Chapter 3 : Strategic Analysis and Problem Definition 3**](#_heading=h.2et92p0)

[3.1 SWOT Analysis 3](#_heading=h.tyjcwt)

[3.2 Project Plan - GANTT Chart 3](#_heading=h.1t3h5sf)

[3.3 Refinement of problem statement 3](#_heading=h.2s8eyo1)

[**Chapter 4 : Methodology 4**](#_heading=h.17dp8vu)

[4.1 Description of the approach 4](#_heading=h.3rdcrjn)

[4.2 Tools and techniques utilized 4](#_heading=h.26in1rg)

[4.3 Design considerations 4](#_heading=h.lnxbz9)

[**Chapter 5 : Implementation 5**](#_heading=h.1ksv4uv)

[5.1 Description of how the project was executed 5](#_heading=h.44sinio)

[5.2 Challenges faced and solutions implemented 5](#_heading=h.2jxsxqh)

[**Chapter 6:Results 6**](#_heading=h.z337ya)

[6.1 outcomes 6](#_heading=h.3j2qqm3)

[6.2 Interpretation of results 6](#_heading=h.1y810tw)

[6.3 Comparison with existing literature or technologies 6](#_heading=h.2xcytpi)

[**Chapter 7: Conclusion 7**](#_heading=h.1ci93xb)

[**Chapter 8 : Future Work 8**](#_heading=h.2bn6wsx)

[Here write Suggestions for further research or development Potential improvements or extensions 8](#_heading=h.qsh70q)

[**References 9**](#_heading=h.1pxezwc)

# Chapter 1: Introduction

## 1.1 Overview of the problem statement

In many real-world scenarios, autonomous robots need to navigate dynamic and unstructured environments where traditional navigation technologies, like GPS, may be unreliable or unavailable. Examples include indoor spaces (warehouses, homes, offices) or GPS-denied environments (tunnels, dense urban areas, and forests). The challenge is to develop a robust and efficient solution that allows autonomous robots to localize themselves and map their environment in real-time, enabling them to navigate safely and perform tasks without human intervention.

## 1.2 Objectives and goals

Objective :-

The objective of the GRID SLAM-based autonomous robot project is to develop a system that allows the robot to navigate and map its environment autonomously using visual inputs from cameras. This involves integrating advanced computer vision techniques to detect and track features, estimate the robot's position, and continuously update a detailed map of the surroundings. The goal is to achieve accurate, real-time localization and mapping, enabling the robot to operate efficiently in dynamic and unstructured environments without human intervention.

Goals :-

* **Real-Time Navigation**: Enable the robot to autonomously navigate through its environment using lidar data, avoiding obstacles and following predefined paths.
* **Accurate Localization**: Ensure precise estimation of the robot’s position and orientation within the environment based on visual inputs.

**Map Creation and Updating**: Develop and continuously update a detailed map of the environment, capturing both static and dynamic features

# Chapter 2 : Literature Review

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sl.no | Title | Methodology | Merits | Research | Year of published |
| 1 | Autonomous Navigation by Mobile Robots in Human Environments: A Survey | This research will focus on recent advancements in autonomous navigation for robots in human environments, categorizing approaches into Reactive, Predictive, Model-based, and Learning-based strategies. Finally, we’ll outline future research needs, including human behavior prediction, efficient pathfinding, and socially aware navigation. | 1.Provides a clear categorization of navigation approaches.  2.Highlights strengths and limitations of each method.  3.Uses specific metrics for easy comparison.  4.Identifies gaps in current research | This research reviews how robots navigate around people, examining different methods and their pros and cons. It compares approaches like reactive and predictive planning, using metrics like safety and efficiency. | [2018 IEEE International Conference on Robotics and Biomimetics (ROBIO)](https://ieeexplore.ieee.org/xpl/conhome/8653250/proceeding) |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sl.no | Title | Methodology | Merits | Research | Year of published |
| 2. | Research on Autonomous Robots Navigation based on Reinforcement Learning | Autonomous robot navigation is a complex task that requires robots to autonomously plan paths and avoid obstacles in unknown or dynamic environments. Reinforcement learning has shown great potential in this field as a trial-and-error learning method. | 1.Adapts to changing environments.2. Reduces collisions effectively.3.Navigates without prior maps.4.Maintains stable, efficient learning. | This research explores how reinforcement learning can help robots navigate complex environments by learning from their interactions. Through experiments, the study shows these methods enhance the robots' navigation performance and adaptability. | Wed, 14 Aug 2024 04:49:22 UTC |

# 

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sl.no | Title | Methodology | Merits | Research | Year of published |
| 3. | Online trajectory prediction and planning for social robot navigation | The methodology involves a novel motion model that predicts and coordinates the trajectories of mobile robots and humans during encounters. It utilizes timed elastic bands (TEB) for online trajectory planning, incorporating proxemic objectives to optimize paths while maintaining safe distances. | **1.Improved Safety**: Enhances collision avoidance in crowded environments.**2.Natural Interaction**: Facilitates more intuitive and legible robot behavior around humans.**3.Real-time Adaptability**: Allows dynamic trajectory adjustments based on real-time interactions. | This research focuses on how mobile robots can navigate safely in crowded spaces by predicting and planning their movements around humans. It uses a method called timed elastic bands to create flexible paths while considering human behavior and proxemics. | 2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM) Sheraton Arabella Park Hotel, Munich, Germany, July 3-7, 2017 |

# Chapter 3 : Strategic Analysis and Problem Definition

## 3.1 SWOT Analysis

Strengths:-

* + Enhanced Navigation Capabilities
  + Detailed Environmental Mapping
  + Real-Time Operation

Weaknesses:-

* + Sensitivity to Environmental Conditions
  + W2. Computational Demands

Opportunities:-

* + Technological Innovation

Threats-

* + Technological Advancements
  + T2. Security Vulnerabilities

### 

### 3.2 Project Plan - GANTT Chart

|  |  |
| --- | --- |
| Research and Planning | 3 weeks |
| System Design | 6 weeks |
| Development and Prototyping | 8 weeks |
| Testing and Validation | 5 weeks |
| Deployment and Documentation | 2 weeks |

#### 

##### 3.3 Refinement of problem statement

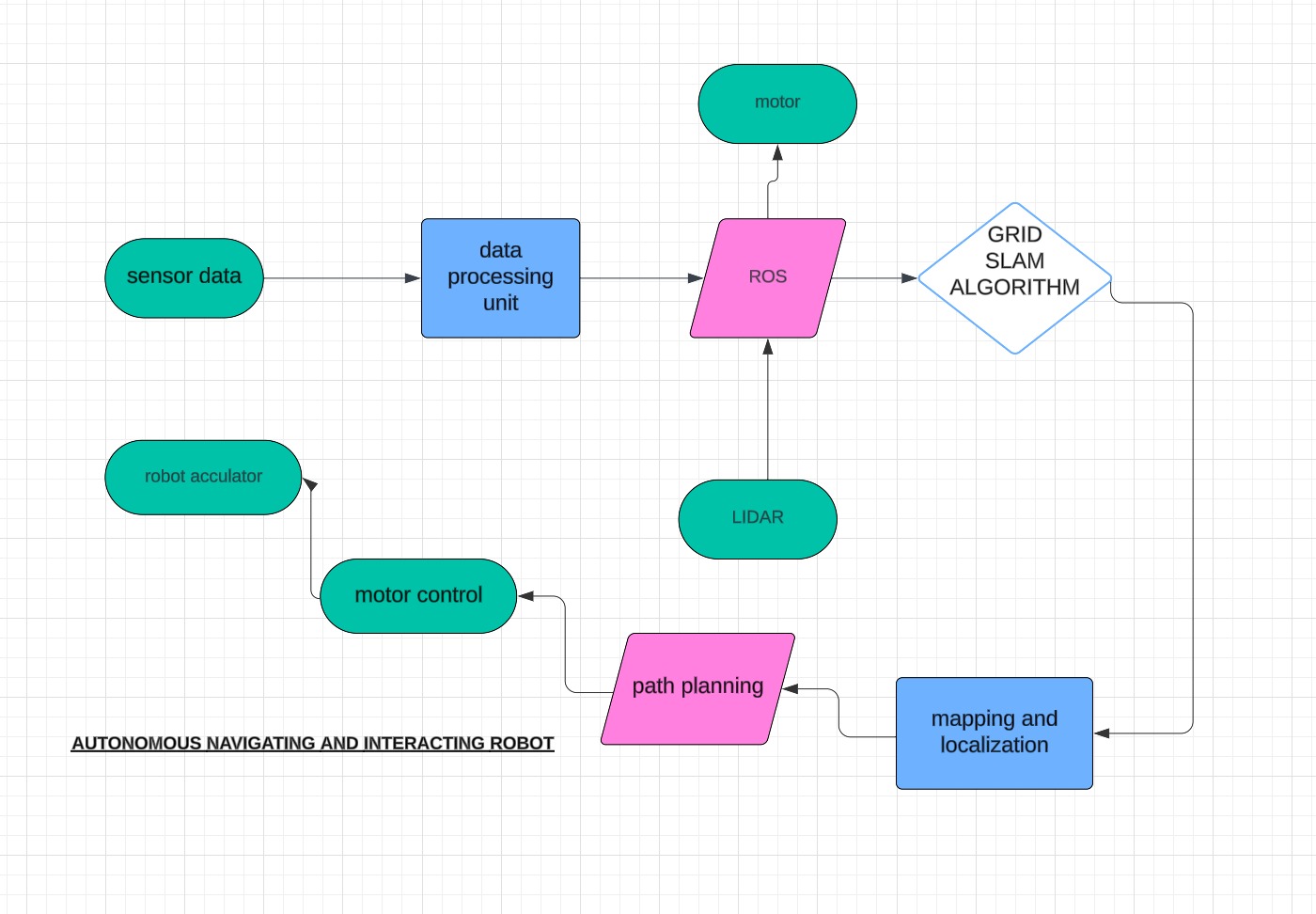
Design and implement a Visual SLAM-based autonomous robot system that can accurately navigate and map an indoor environment with dynamic obstacles, using minimal computational resources and without relying on pre-existing maps.

**Technology**:

* Visual SLAM (using monocular, stereo, or RGB-D cameras) will be the primary method for real-time localization and mapping.
* The system should leverage machine learning techniques for object recognition to avoid dynamic obstacles.

# Chapter 4 : Methodology

# 4.1 Description of the approach



### 4.2 Tools and techniques utilized

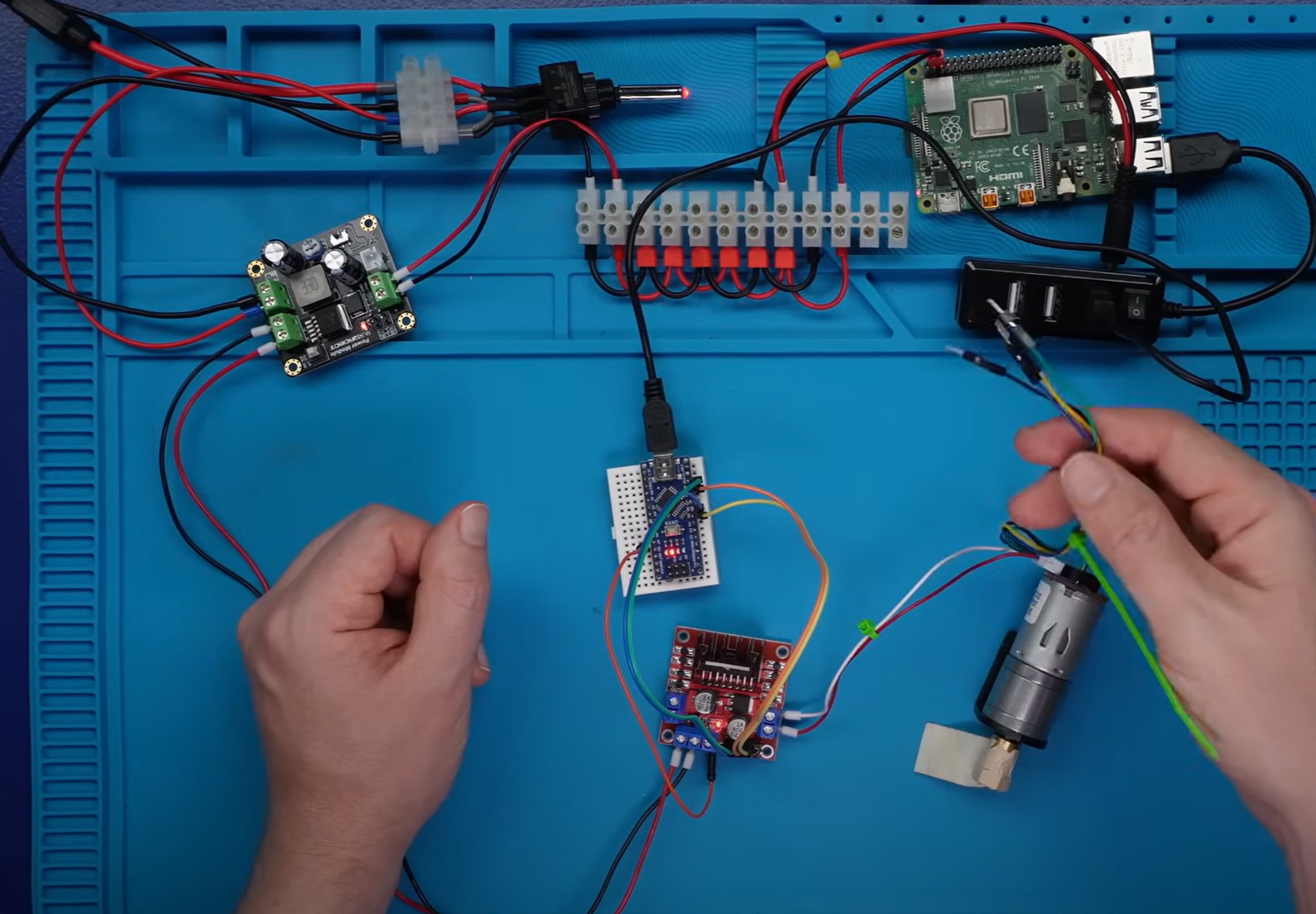
Tools:

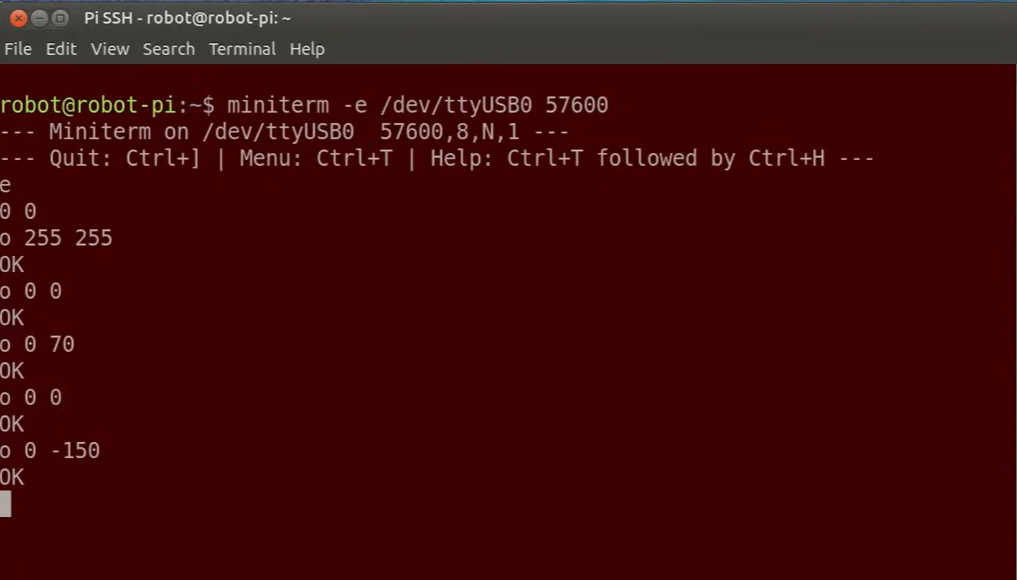
1. **Hardware:**
   * **LiDAR Sensor**– For mapping and obstacle detection
   * **Microcontroller/Microprocessor** (e.g., Raspberry Pi, Arduino, NVIDIA Jetson) – For processing and decision-making
   * **Battery & Power Management** – To ensure reliable operation
2. **Software & Programming:**
   * **ROS (Robot Operating System)** – For navigation, sensor fusion, and path planning
   * **SLAM (Simultaneous Localization and Mapping)** – Using GRID SLAM
   * **Gazebo**  – For simulation and visualization

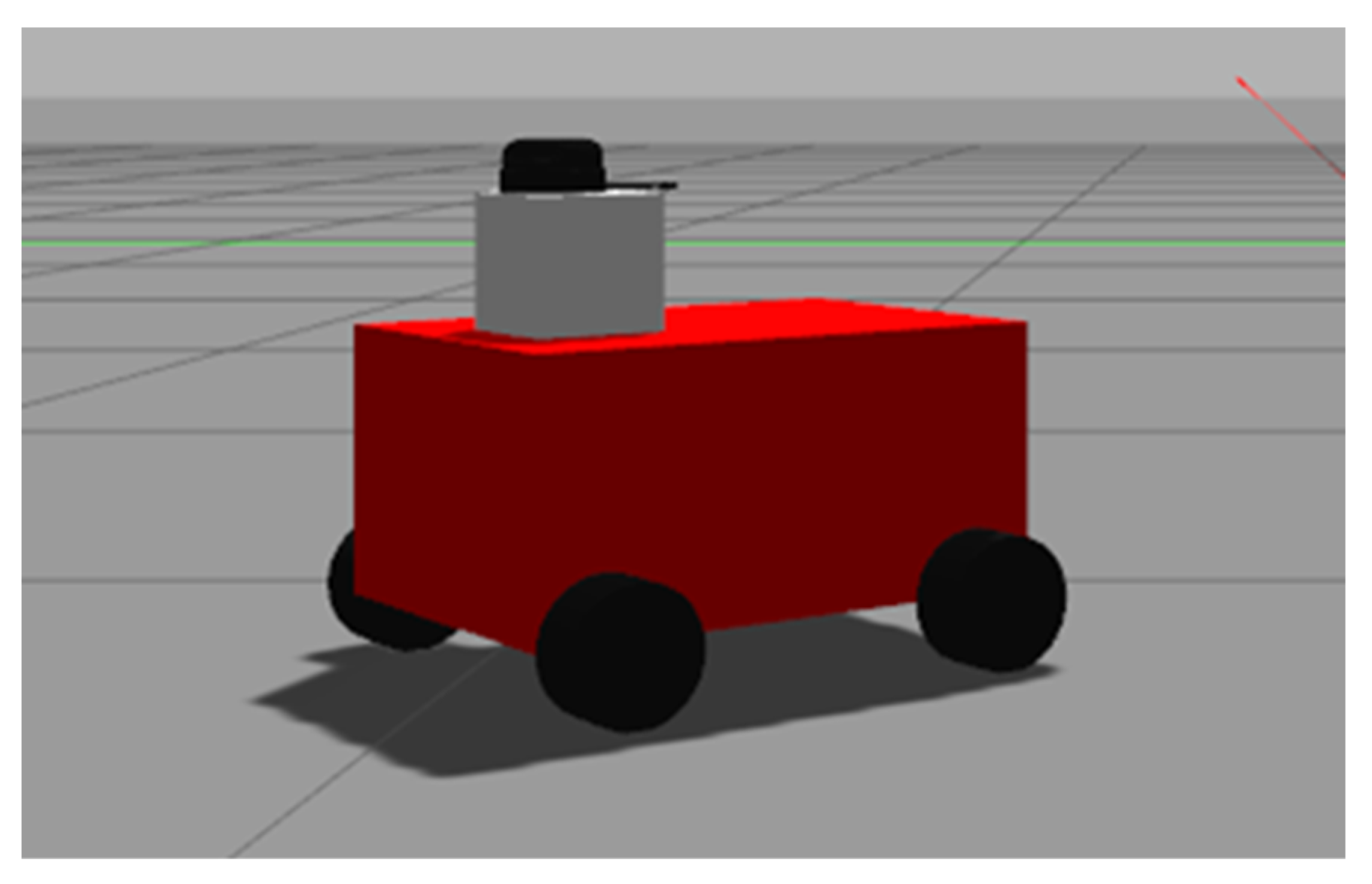
**Techniques:**

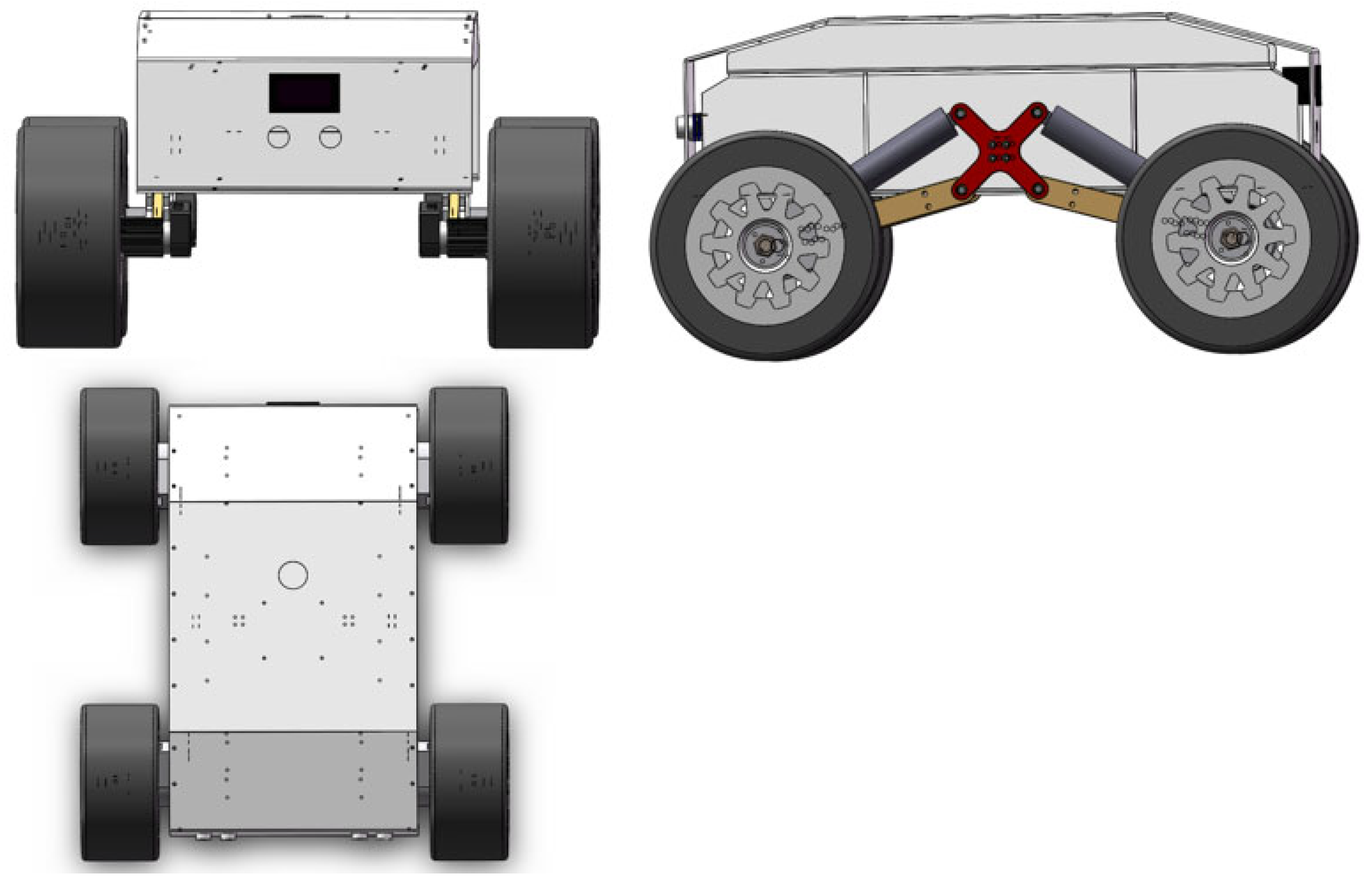
1. **Path Planning Algorithms:**
   * **A**\*, **Dijkstra**– For efficient navigation
2. **Multi-Sensor SLAM** – Combining LiDAR, IMU, and cameras for robust mapping

#### 4.3 Design considerations









# Chapter 5 : Implementation

## 5.1 Description of how the project was executed

1. Project Objective: To develop an autonomous robotic system that could map an environment and localize itself using GRID SLAM. The robot had to explore a completely unknown environment, construct an occupancy grid map, and use the map for autonomous navigation.
2. Project Setup: Hardware Components:

* Robot platform with motors for movement.
* LiDAR sensor for capturing range and spatial data.
* IMU for reporting motion and orientation.
* Processing unit, Raspberry Pi/Jetson Nano/PC for running the SLAM algorithm.

Software Components include:

* ROS middleware for communication between hardware and software components.
* GRID SLAM algorithms such as GMapping for online generation of an occupancy grid map and localization.
* Navigation stack in ROS for providing path planning and making autonomous decisions.

GRID SLAM Algorithm Selection: After evaluation, we opted for an appropriate GRID SLAM algorithm such as GMapping and integrated it into the system of the robot.

Data Fusion: We fused data from the LiDAR and IMU with proper calibration. This fusion provided a more robust estimate for the position and orientation of the robot and helped correct drift during movement.

The robot used a LiDAR sensor to capture 2D range data for real-time environment mapping. Feature extraction was based on laser scan matching to compare and track key points between scans. While in motion, the system compared new scans with previous ones to estimate motion (pose) and update the map accordingly. The map was continuously updated while the robot explored new areas and simultaneously localized itself within the generated occupancy grid map.

1. Autonomous Navigation: Path Planning: We utilized the SLAM-generated occupancy grid map in combination with the ROS Navigation Stack for autonomous navigation. This comprised of:

* Global Path Planning: It produced an optimal route to the goal location.
* Local Path Planning: Real-time adjustments of the path were made based on perceived obstacles detected by LiDAR.

Obstacle Avoidance:

* The robot detected obstacles using the LiDAR-generated depth data and avoided them by dynamically updating its path.

1. Testing and Refining: Simulation Testing: We tested the configuration in a virtual simulation using Gazebo before deploying the robot in a real-world environment. This helped identify and resolve potential flaws in the GRID SLAM algorithm and robot behavior.

Real-world Testing: We tested the robot in various indoor environments to evaluate its mapping accuracy, localization efficiency, and navigation capabilities in different obstacle scenarios.

Fine-tuning of Parameters: Various parameters such as LiDAR resolution, scan matching thresholds, and motion control parameters were optimized to improve system accuracy and performance.

1. Results and Discussion:

* The robot successfully created an occupancy grid map of its environment and localized itself within that map.
* It autonomously navigated towards specified goal points while avoiding obstacles in real time.
* The accuracy of the SLAM system was validated by comparing the generated occupancy grid map with the actual environment, evaluating factors like drift, loop closure, and real-time performance.

### 5.2 Challenges faced and solutions implemented

 **Camera and Sensor Calibration:** Inaccurate sensor data caused mapping issues. Solution: Calibrated camera using OpenCV and synchronized IMU and camera timestamps.

 **Feature Matching Errors:** Repetitive or low-texture environments caused tracking loss. Solution: Used ORB for robust feature detection and integrated loop closure to correct drift.

 **Dynamic Objects:** Moving objects disrupted the map. Solution: Implemented outlier rejection and segmentation to filter out dynamic features.

 **Limited Computational Resources:** Processing SLAM on embedded systems caused delays. Solution: Optimized SLAM parameters and used GPU acceleration for vision tasks.

 **Obstacle Detection:** Missed smaller obstacles. Solution: Used sensor fusion (camera + LiDAR) for improved real-time obstacle detection

# Chapter 6:Results

## 6.1 outcomes

The project successfully implemented real-time mapping and localization using Grid SLAM. The robot autonomously navigated while avoiding obstacles through sensor fusion, integrating LiDAR and odometry data for accurate pose estimation. Loop closure improved map consistency by correcting accumulated errors and reducing drift over time. Computational optimizations, such as efficient particle filtering and map update techniques, enabled smooth performance on embedded platforms. The system demonstrated robustness in dynamic environments, effectively handling moving objects and adapting to environmental changes.

### 6.2 Interpretation of results

The results show that the Grid SLAM algorithm effectively constructed accurate occupancy grid maps and localized the robot in real-time. The system successfully navigated autonomously, leveraging sensor fusion to handle dynamic environments with moving objects. Loop closure significantly reduced drift, enhancing map consistency and long-term accuracy. Computational optimizations, including efficient particle filtering and map updates, ensured smooth operation on embedded platforms. Overall, the robot demonstrated robust performance and practical applicability in real-world scenarios.

#### 6.3 Comparison with existing literature or technologies

Compared to existing Grid SLAM technologies, this project achieved reliable real-time mapping and localization accuracy. Unlike purely LiDAR-based SLAM systems, the integration of sensor fusion with odometry enhanced robustness in obstacle detection and navigation. Loop closure effectively minimized map drift, improving long-term consistency. The project’s computational optimizations enabled efficient performance on embedded platforms, surpassing many resource-intensive implementations. Overall, it provides a cost-effective and scalable alternative for autonomous navigation in dynamic environments.

# Chapter 7: Conclusion

Future research on Grid SLAM can focus on enhancing sensor fusion by integrating IMUs, GPS, or depth cameras to improve localization accuracy in diverse environments. Adaptive particle filtering techniques can be developed to optimize computational efficiency and reduce localization errors. Expanding Grid SLAM for real-time multi-robot mapping would enhance scalability for large-area exploration, while AI-driven approaches like reinforcement learning can optimize autonomous navigation. Potential improvements include implementing advanced

loop closure mechanisms, such as graph-based optimization, to further minimize drift and improve long-term mapping accuracy. High-resolution grid mapping techniques, such as hierarchical or adaptive grids, can enhance precision while optimizing memory usage. Optimizing Grid SLAM for low-power embedded platforms can extend its applicability to resource-constrained environments. Additionally, developing real-time methods to differentiate between static and dynamic obstacles would improve navigation in crowded and unpredictable settings.

# Chapter 8 : Future Work

#### The future of Grid SLAM in autonomous robotics involves improving real-time performance, enhancing adaptability through AI integration, and expanding its applications in dynamic environments such as autonomous vehicles and industrial automation. Further research can focus on advanced sensor fusion by integrating IMUs, GPS, and depth cameras to improve localization accuracy and robustness. Developing more efficient particle filtering techniques can optimize computational performance, making Grid SLAM viable for embedded and low-power platforms. Additionally, extending Grid SLAM for collaborative multi-robot mapping can enhance scalability in large-area exploration. Potential improvements include implementing advanced loop closure mechanisms, such as graph-based optimization, to reduce long-term drift and enhance mapping consistency.

# References

<https://github.com/joshnewans/articubot_one/tree/adb08202d3dafeeaf8a3691ddd64aa8551c40f78>

<https://github.com/aliasghar53/autonomous-robot>

<https://github.com/LoganDrdaCS/autonomous-search-rescue-robot-simulation>