



Faculty of Engineering and Technology
Department of Robotics and automation Engineering
Jain Global Campus, Kanakapura Taluk - 562112
Ramanagara District, Karnataka, India

2019-2023

A Project Report on

“Ditto: The delivery robot”

Submitted in partial fulfilment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

ROBOTICS AND AUTOMATION

Submitted by

Name: Dennis McLeaord.A
USN: 19BTRRA003

Name: Athish Anand Kumar
USN: 19BTRRA005

Name: Chevula Haarvish
USN:19BTRRA009

Name: Siddarth.D
USN:19BTRRA015

Under the guidance of
Mrs.Mamatha GN

Assistant Professor

Department of Electronics and Communication Engineering Faculty of
Engineering & Technology

JAIN DEEMED-TO-BE UNIVERSITY

Faculty of Engineering & Technology

Department of Robotics and automation Engineering

Jain Global campus
Kanakapura Taluk - 562112
Ramanagara District
Karnataka, India

CERTIFICATE

This is to certify that the project work titled “DITTO:THE DELIVERY ROBOT” is carried out by **Dennis McLeaord.A (19BTRRA003), Athish Anand Kumar (19BTRRA005), Chevula Haarvish(19BTRRA009), Siddarth.D (19BTRRA015)** are bonafide students of Bachelor of Technology at the Faculty of Engineering & Technology, JAIN DEEMED-TO-BE UNIVERSITY, Bengaluru in partial fulfillment for the award of degree in Bachelor of Technology in Electronics and Communication Engineering, during the academic year **2022-2023**.

Prof Mamatha GN

Assistant Professor
Dept. of ECE,
Faculty of Engineering &
Technology,
JAIN DEEMED-TO-BE
UNIVERSITY
Date:

Dr. R. Sukumar

Head of the Department,
ECE,
Faculty of Engineering &
Technology,
JAIN DEEMED-TO-BE
UNIVERSITY
Date:

Dr. Hariprasad S.A

Professor/Director,
Faculty of Engineering &
Technology,
JAIN DEEMED-TO-BE
UNIVERSITY
Date:

Name of the Examiner

Signature of Examiner

1.

2.

DECLARATION

We, **Dennis McLeaord.A (19BTRRA003)**, **Athish Anand Kumar (19BTRRA005)**, **Chevula Haarvish (19BTRRA009)**, **Siddarth.D (19BTRRA015)**, are students of eighth semester B.Tech in **Robotics And Automation Engineering**, at Faculty of Engineering & Technology, **JAIN DEEMED-TO-BE UNIVERSITY**, hereby declare that the project titled "**Ditto-The Delivery Robot**" has been carried out by us and submitted in partial fulfilment for the award of degree in **Bachelor of Technology in Robotics And Automation Engineering** during the academic year **2022-2023**. Further, the matter presented in the project has not been submitted previously by anybody for the award of any degree or any diploma to any other University, to the best of our knowledge and faith.

Signature

Dennis McLeaord.A

(19BTRRA003)

Athish Anand Kumar

(19BTRRA005)

Chevula Haarvish

(19BTRRA009)

Siddarth.D

(19BTRRA015)

Place : Bengaluru

Date :

ACKNOWLEDGEMENT

It is a great pleasure for us to acknowledge the assistance and support of a large number of individuals who have been responsible for the successful completion of this project work.

First, we take this opportunity to express our sincere gratitude to **Faculty of Engineering & Technology, JAIN DEEMED-TO-BE UNIVERSITY** for providing us with a great opportunity to pursue our Bachelor's Degree in this institution.

In particular we would like to thank **Dr. Hariprasad S.A, Director, Faculty of Engineering & Technology, JAIN DEEMED-TO-BE UNIVERSITY** for his constant encouragement and expert advice.

It is a matter of immense pleasure to express our sincere thanks to **Dr. R. Sukumar, Head of the department, Electronics and communication Engineering, JAIN DEEMED-TO-BE UNIVERSITY**, for providing right academic guidance that made our task possible.

We would like to thank our guide **prof Mamatha GN, Assistant Professor, Dept. of Electronics and Communication Engineering, JAIN DEEMED-TO-BE UNIVERSITY**, for sparing his/her valuable time to extend help in every step of our project work, which paved the way for smooth progress and fruitful culmination of the project.

We would like to thank our Project Coordinator **Mr. Sunil M P** and all the staff members of Electronics and Communication for their support.

We are also grateful to our family and friends who provided us with every requirement throughout the course.

We would like to thank one and all who directly or indirectly helped us in completing the Project work successfully.

Signature of Students

TABLE OF CONTENTS

List of Figures	7
List of Tables	7
Nomenclature used	8
Chapter 1	10
1. INTRODUCTION	10
1.1 Literature Survey	12
1.2 Challenges and problems	23
1.3 Objectives	24
1.4 Methodology	24
1.5 Hardware and software tools uses	30
Chapter 2	35
2. BASIC THEORY BEHIND WORKING OF DITTO	35
Chapter 3	35
3. TOOL DESCRIPTION	35
Chapter 4	40
4. IMPLEMENTATION	40
4.1 Hardware Design and Implementation	40
4.2 Building prototype	41
4.3 Software used	42
4.4 Techniques used in localization & Navigation Layer	46
4.5 Delivery Stack	60
Chapter 5	63
Result and Discussions	63
CONCLUSIONS AND FUTURE SCOPE	64
REFERENCES	65

Appendix -I	67
Information regarding Students	68
BATCH PHOTOGRAPH ALONG WITH GUIDE	69

Fig. No.	Description of figure	Page No.
3.1	Proposed Block Diagram	23
3.1(a)	Raspberry Pi4	24
3.1(b)	Temperature Sensor (DHT11)	25
3.1(c)	Ultrasonic Sensor	26
3.1(d)	OAK-D Lite Depth Camera	27
4.0	Ditto CAD model	28
4.2.1	Pre-Body work of the Robot	29
4.2.2	Final Output of the Robot	30
4.3.2(1)	Fusion 360 model of the Robot	30
4.3.2(2)	Gazebo Model	32
4.3.2(3)	Gazebo Model	32
4.3.2(4)	Gazebo Model	33
4.4.1(1)	Gazebo Model	33
4.4.1(2)	Road Segmentation	37
4.4.2(3)	Color Based Segmentation	37
4.4.2(1)	Object Detection	38
4.4.2(2)	Localization	41
4.4.2(3)	Obstacle Detection	42
4.4.2(4)	Package Detection	43
4.4.2(5)	Delivery Point Identification	43
4.4.2(6)	Traffic Signal Identification	44
4.4.3(1)	Security Anomaly Detection	44
4.4.3(2)	Localization Mapping	45
4.4.3(3)	LIDAR Based Localization	46

NOMENCLATURE USED

GPS	Global positioning sensing
GUI	Graphical User Interface
USB	Carriage Return
ROS	Robot Operating System
OAK-D Lite	OAK-D Lite Depth Camera
EKF	Extended Kalman Filter

ABSTRACT

E-commerce and package deliveries are expanding quickly, and numerous start-ups have already started testing autonomous delivery robots to bring goods and groceries to customers (ADRs). The fast advancement of robot technology has made food delivery robots a focus of both domestic and international studies. In this study, a food delivery robot was developed based on the integration of Lidar and machine vision, which combined the information of Lidar and machine vision in time and location. Particularly, their data may be unified in time due to the synchronization of Lidar and machine vision in working frequency. Additionally, certain pertinent data can be unified in space by the equivalent conversion of the radar coordinate system, the world coordinate system, the picture coordinate system, and the pixel coordinate system.

In this report, the robotic food delivery system makes use of the server for data processing and data transfer. The originality of this study is the integration machine vision to provide the robot with precise environmental awareness. According to the experimental findings, combining data from several sensors can significantly increase the robot's efficiency at delivering meals by lowering coordinate cumulative errors and the likelihood of "Suspended animation" during operation. The robot also features accurate obstacle avoidance, path planning, and autonomous navigation capabilities.

CHAPTER 1

INTRODUCTION:

The field of autonomous robots is rapidly expanding, with a diverse range of emerging applications and growing interest from various industries such as automotive, trucking, public transportation, industrial, and military. These robots have the potential to significantly improve safety and operational efficiency. With the rise of e-commerce, faster, more affordable, and sustainable last-mile deliveries have become increasingly important, and autonomous robots can help solve challenges such as reduced capacity, driver shortage, damaged and stolen products, failed delivery attempts, and traffic congestion. Autonomous robots are designed to operate independently and efficiently, making quick decisions even in adverse conditions while ensuring the safety of pedestrians. The ultimate goal is for autonomous robots to work in harmony with humans, making their lives easier.

Autonomous robots are now widely used in various industries, households, military applications, and disaster management across the globe. These robots provide a safer and more



Efficient alternative to perform tasks that are difficult or time-consuming for humans. Examples of autonomous robots include Roomba cleaning robots, delivery robots, and autonomous vehicles that move freely in physical spaces without human guidance. To achieve complete autonomy, a robot must be fully aware of its surroundings and capable of taking action based on the inputs received from various modules. This requires the robot to obtain information from sensors, perceive the environment, precisely locate itself in the world, and create an optimal plan to achieve its goal. The robot then integrates the instructions obtained from these modules in real-time and sends them to a control node to move the system in the real world.

A robot must have a comprehensive awareness of its environment and the ability to operate in response to inputs received through multiple system modules in order

to be fully autonomous. To reach a state of total autonomy, a robot needs to be able to gather data from sensors, comprehend its surroundings, locate itself precisely in space, and finally come up with the best strategy for attaining its objective. To actually move the system in the real environment, the robot must combine the instructions obtained from the aforementioned modules in real time and send them to a control node. For an autonomous robot to function, the accuracy and effective integration of all the modules is crucial. Any one of the modules could have a significant flaw that could even endanger people nearby.

How Do Delivery Robots Get Around?

When you enter the locations into navigation software, the path between a vendor and a delivery site may appear A-to-B but there are added considerations for a delivery robot, such as sidewalks, crossings, driveways, individuals, animals, automobiles, and so on. The robots on Star ship compute a path based on the shortest distance and satellite pictures of the route. Each route feature (crossings, driveways, etc.) receives a time computation, which the robot uses to determine the best path and delivery time. Over time, the robots create a wireframe map of continuous elements (buildings, crossings, statues, paths, etc.) and ensure that future travels through the region are faster. Is there somebody in charge of the delivery robot? While the Star ship Technology robots are self-contained, they are not cut off from their humans. If a robot encounters a substantial difficulty, such as a very large curb (they can climb up and over typical sidewalk curbs), a human operator can take over and solve the problem.

To summarize, delivery robots are a promising solution to meet the increasing demand for convenient and efficient last-mile delivery services. This report has provided an overview of delivery robots, their features, applications, and potential benefits. Various types of delivery robots, including ground-based autonomous vehicles and aerial drones, offer unique advantages and challenges. The adoption of delivery robots by major companies and logistics providers indicates their potential to revolutionize the delivery industry. By utilizing advanced technologies like artificial intelligence, machine learning, and sensors, delivery robots can navigate complex environments, ensure the safety of goods, and optimize delivery routes. Furthermore, delivery robots offer environmental advantages by reducing carbon emissions and traffic congestion, aligning with the growing focus on sustainability.

1.1 Literature Survey

[1] **Alexander Buchegger et.al, 2018.** To allow autonomous transport vehicles to be used for transportation tasks in large-scale outdoor environments proven approaches from the robotics domain needs to be applied and transferred to these new environments. In this paper, we present an integrated autonomous transport vehicle which addresses these problems and is able to deliver parcels in urban environments such as city centers automatically. The developed transport vehicle is based on a commercial electrical personal vehicle. It was adapted for autonomous control and equipped with improved navigation skills for outdoor environments based on a topological navigation approach. The integrated vehicle was evaluated in realistic delivery use cases where parcels are delivered autonomously to addresses in a larger urban area. The main contributions of this paper are: (1) the adaptation of well-known algorithms for robot navigation for large-scale urban environments, (2) an integration of these algorithms in a commercially available electrical vehicle, (3) the improvement of the robustness of the approach by integrating additional from OpenStreetMap (OSM), and (4) an evaluation of the autonomous delivery concept in real urban environments such as an university campus or a city center.

[2] **Aniket Gujarathi et.al, 2019** The field of autonomous robots is growing rapidly in the world, in terms of both the diversity of emerging applications and the levels of interest among traditional players in the automotive, truck, public transportation, industrial, and military communities. Autonomous robotic systems offer the potential for significant enhancements in safety and operational efficiency. Due to the meteoric growth of e-commerce, developing faster, more affordable and sustainable last-mile deliveries become more important. In this paper, Autonomous robot including the cyber physical architecture of the robots as well as the renderings of CAD models are illustrated. Designing new solutions including catadioptric cameras that output panoramic views of the scene, i.e., images with very large fields of view. It describes the problem of state estimation and localization of a robot in detail. In order to navigate accurately around the world, the robot must know its location in the world and the map exactly. A robot can move smoothly only if it is properly localized. An inaccurate localization may cause the robot to vary on the roads or behave erroneously which are serious issues when the robot is completely autonomous.

[3] **Dae-Nyeon Kim et.al 2018**, When an autonomous robot navigates, it is likely for him to set specific a target. This paper focuses on object recognition. He also needs avoid objects when he encounters obstacle, and

Know where he is and know further path take, he. To recognize an object, we classify object into artificial and natural. Then we define their characteristics individually. We segment the object after the process of preprocessing. Image segmentation delineates boundaries between meaningful components, while object recognition attempts to find instances of objects within an image. We propose a method to segment objects of outdoor environment using multiple features. To analyses and recognize specific object, our method used property of segmented objects. This paper proposed the method object recognition of outdoor environment using segmented region by multiple features. The PCs are used to recognize the building. The meshes of parallelograms can help us to detect more. In addition, the relation of geometrical properties as the height and the number of windows can be exploited to analyze more information of building. For example, how many rooms the building has. This process is preprocessing objects from an image taken by moving robot in an outdoor environment.

[4] **Murad Mehrab Abrar et.al, 2021** Robots and autonomous vehicles can help to ease the stress on the existing home delivery while reducing the risk of virus transmission by mitigating direct human contact. In this regard, we have developed a cost effective autonomous mobile robot prototype for the purpose of increasing the last mile delivery efficiency as well as ensuring a secure and contactless package delivery. An autonomous mobile robot is a self-driving vehicle that does not require any operation from operator to navigate the robot. The movements and trajectory are predefined before the operation and the robot navigates accordingly. Among various navigation techniques, we have used the Global Positioning System (GPS) data for autonomous navigation of the robot and the destination is predefined as latitude and longitude points in the program of the robot. The main advantage of using GPSS for navigation is that the data received from the GPS are independent of the previous readings; therefore, it is easy to minimize errors. A digital compass measures the heading angle of the robot and helps the robot to find the direction of the trajectory. The robot is equipped with a password protected container which protects the package against theft, damage and unprotected human contact. This password can be sent to the customer by a text message from the service company. Once the robot arrives at its delivery location, the only person who has the password will be able to unlock its delivery.

[5] **Nalinaksh Vyas et.al 2019**. In the paper "Delivery Robots in Logistics: A Review of Recent Advances and Challenges" by Nalinaksh Vyas and Arindam Ghosh (2021), the authors provide a thorough examination of the latest developments and obstacles concerning delivery robots in the field of logistics. They emphasize the significance of last-mile delivery and how delivery robots can contribute to overcoming the associated difficulties in the supply chain. The authors discuss different types of delivery robots, such as ground-based robots, aerial drones, and autonomous vehicles, outlining their capabilities, limitations, and practical applications. They also explore the technological progress made in robot perception, navigation, and manipulation, which has significantly improved the performance and feasibility of delivery robots. The integration of these robots with existing logistics infrastructure and the role of emerging technologies like artificial intelligence and machine learning in enhancing their efficiency are also discussed. The second part of the paper concentrates on the challenges and considerations involved in deploying delivery robots in real-world logistics operations. The authors examine regulatory and legal concerns, safety issues, social acceptance, and economic viability. They analyse the potential impact of delivery robots on employment and stress the need for collaboration between humans and robots in logistics operations. The paper further highlights operational challenges such as route planning, battery life, payload capacity, and scalability that need to be addressed. Additionally, the authors discuss the environmental implications of employing delivery robots and emphasize the importance of sustainable solutions. Overall, the paper provides an extensive overview of recent advancements in delivery robots while identifying key challenges that must be overcome for successful integration into the logistics industry.

[6] **Vikas Kumar et.al,2021** The paper focuses on the design and development of a cost-effective autonomous delivery robot specifically tailored for indoor environments. The authors present their approach to constructing a robot that can autonomously navigate indoor spaces and deliver items efficiently. They begin by discussing the motivation behind the project, highlighting the increasing demand for automated delivery systems in various sectors such as hospitals, warehouses, and offices. The authors emphasize the need for a low-cost solution to make the technology accessible to a wider range of organizations. The paper outlines the design considerations for the robot, which include its physical structure, locomotion mechanism, and sensing capabilities. The authors describe the use of a

differential drive system to enable the robot to manoeuvre through narrow spaces effectively. They also discuss the integration of sensors, such as ultrasonic sensors and a laser scanner, to facilitate obstacle detection and mapping of the robot's environment. The authors detail the development of the robot's autonomous navigation system, which involves the use of localization algorithms and path planning techniques. They explain how the robot utilizes a combination of odometry and sensor data fusion to estimate its position accurately. Additionally, the paper highlights the implementation of a map-based navigation approach that enables the robot to plan optimal paths for item delivery within the indoor environment. The authors present the results of experimental evaluations conducted to assess the performance of the autonomous delivery robot. They discuss the robot's ability to navigate through complex indoor spaces, avoid obstacles, and successfully deliver items to designated locations. The paper concludes with a discussion of the practical applications and potential future improvements for the low-cost autonomous delivery robot. Overall, the paper provides a detailed account of the design, development, and evaluation of a low-cost autonomous delivery robot tailored for indoor environments. It showcases the successful implementation of the robot's physical structure, sensing capabilities, and navigation system, highlighting its potential to address the increasing demand for automated delivery systems in various indoor settings.

[7] **Kumar et al., 2019** In their paper, Kumar, Moreira, and Schillers provide a comprehensive review of delivery robots for last-mile logistics operations. They emphasize the significance of the last-mile phase in logistics and the challenges associated with it. The authors discuss the increasing interest in delivery robots as a potential solution to address these challenges, offering benefits such as improved efficiency, reduced costs, and enhanced sustainability.

The paper explores various types of delivery robots, including ground-based robots, aerial drones, and autonomous vehicles. Kumar et al. examine the key features, functionalities, and technological aspects of these robots, such as perception, navigation, manipulation, and communication systems. They highlight the advancements in robot hardware and software that have contributed to their increased capabilities and adaptability in real-world logistics scenarios. Furthermore, the authors discuss the operational considerations of deploying delivery robots, including route planning, fleet coordination, load capacity, and safety regulations. They delve into the integration of delivery robots with existing logistics infrastructure, examining the challenges of interoperability and the need for standardized

interfaces. Kumar et al. also addresses social acceptance and public perception of delivery robots, discussing factors such as privacy, security, and the impact on employment.

The paper concludes by identifying research gaps and potential future developments in the field of delivery robots for last-mile logistics. The authors emphasize the need for further advancements in areas such as artificial intelligence, sensing technologies, and human-robot interaction to enhance the capabilities and acceptance of these robots. They highlight the importance of interdisciplinary collaborations and partnerships between academia, industry, and policymakers to foster the successful implementation of delivery robots in last-mile logistics operations. In summary, Kumar, Moreira, and Scholler's review paper provides a comprehensive overview of delivery robots for last-mile logistics operations. It explores various types of robots, their features, and technological aspects. The authors discuss operational considerations, integration challenges, and social acceptance factors. The paper identifies research gaps and emphasizes the need for future advancements and collaborations in the field.

[8] Zhang et al. 2021 In their paper, Zhang, Li, and Zhang provide a comprehensive examination of the application of drones in logistics and supply chain management. The authors highlight the increasing interest in drones as a disruptive technology that has the potential to revolutionize the logistics industry by enabling efficient and cost-effective delivery operations.

The paper begins by discussing the benefits of using drones in logistics, such as reduced delivery time, improved delivery accuracy, and enhanced accessibility to remote areas. Zhang et al. explore different types of drones used in logistics, including fixed-wing drones, multi-rotor drones, and hybrid drones, detailing their capabilities, limitations, and suitability for various applications. The authors delve into the key aspects of drone logistics, including route planning, payload capacity, and safety considerations. They discuss the use of optimization algorithms for route planning, considering factors such as traffic congestion, weather conditions, and energy consumption. Additionally, they examine payload capacity and highlight the challenges of carrying different types of goods, ranging from small packages to heavier cargo. Furthermore, the paper addresses the integration of drones into the existing supply chain management systems. Zhang et al. discuss the potential of using drones for inventory management, warehouse operations, and order fulfilment. They highlight the use of drone-based systems for inventory tracking, real-time monitoring of warehouse operations, and picking and sorting tasks. The authors also discuss the regulatory

and legal challenges associated with drone operations in logistics. They examine the current regulations and restrictions imposed by aviation authorities and highlight the need for standardized rules and guidelines to ensure safe and efficient drone operations. In conclusion, Zhang, Li, and Zhang's comprehensive study provides a detailed overview of the application of drones in logistics and supply chain management. The paper covers various aspects, including drone types, route planning, payload capacity, integration with supply chain management, and regulatory challenges. The authors emphasize the potential benefits and future prospects of using drones in the logistics industry, while acknowledging the need for addressing operational and regulatory considerations to fully unlock their potential.

[9], **Nanda Kumar et.al., 2019** In their paper, Nanda Kumar, Hanumanthappa, and Vijaya Kumar provide a comprehensive review of the use of autonomous unmanned aerial vehicles (UAVs) for delivery services. They discuss the growing interest in UAVs as a potential solution for last-mile delivery, offering advantages such as faster delivery times, reduced costs, and accessibility to remote areas. The authors explore the various aspects of autonomous UAV delivery systems, including UAV design, flight control, navigation, and payload capacity. They discuss the advancements in UAV technologies, such as lightweight materials, energy-efficient propulsion systems, and sophisticated flight control algorithms. Additionally, they delve into the integration of sensing and perception systems, which enable UAVs to detect and avoid obstacles during autonomous flight. Furthermore, the paper examines the regulatory and operational challenges associated with UAV delivery services. Nanda Kumar et al. discuss airspace regulations, safety considerations, and the need for reliable communication systems. They highlight the importance of developing robust UAV control algorithms and ensuring secure data transmission for successful and safe delivery operations. The authors also address the potential applications and benefits of UAV delivery services in various sectors, including e-commerce, healthcare, disaster management, and agriculture. They discuss the use of UAVs in medical supply delivery, emergency response, and precision agriculture, showcasing the potential transformative impact of UAVs on these industries. In conclusion, Nanda Kumar, Hanumanthappa, and Vijaya Kumar's review paper provides a comprehensive overview of the use of autonomous UAVs for delivery services. It covers various aspects, including UAV design, flight control, navigation, regulations, and applications. The authors highlight the advancements in technology, the challenges that need to be addressed, and the potential benefits of UAV delivery systems across different sectors.

Title	Year	Authors	Results
[1]“An Autonomous Vehicle for Parcel Delivery in Urban Areas”	2018	Alexander Buchegger, Konstantin Lassnig, Stefan Loigge, Clemens Muhlbacher, and Gerald Steinbauer,	Autonomous Vehicle Design: The report focuses on the development of an autonomous vehicle specifically designed for parcel delivery in urban areas. Intelligent Routing and Planning: The report discusses the intelligent routing and planning algorithms employed by the autonomous vehicle to optimize parcel delivery.
[2]“Design and Development of Autonomous Delivery Robot	2020	” Aniket Gujarathi, Akshay Kulkarni, Unmesh Patil, Yogesh Phalak, Rajeshree Deotalu, Aman Jain, Navid Panchi under the guidance of Dr. Ashwin Dhabale and Dr. Shital S. Chiddarwar	Robot Design and Development: The paper focuses on the design and development of an autonomous delivery robot. The authors describe the physical design of the robot, including its size, structure, and mobility mechanisms. Navigation and Path Planning: The authors present the navigation and path planning algorithms implemented

			in the autonomous delivery robot. They discuss the use of Simultaneous Localization and Mapping (SLAM) techniques to enable the robot to build a map of its environment and localize itself within it.
[3]“Object Recognition of Outdoor Environment by Segmented Regions for Robot Navigation”	2019	Dae-Nyeon Kim, Hoang-Hon Trinh, and Kang-Hyun Jo,	Segmented Region Object Recognition: The paper focuses on the development of an object recognition system for outdoor environments using segmented regions. The authors propose a method that segments the captured image of the environment into different regions based on color and texture information. Robot Navigation and Object Awareness: The authors present how the object recognition system is utilized for robot navigation and object awareness. The recognized objects in the environment are utilized to generate a map or

			model of the surroundings, enabling the robot to understand its environment better.
[4]"An Autonomous Delivery Robot to Prevent the Spread of Coronavirus in Product Delivery System"	2021	Murad Mehrab Abrar, Raian Islam, Md. Abid Hasan Shanto, Ahsanullah	Contactless Delivery System: The authors present the implementation of a contactless delivery system in the autonomous robot. The system ensures the safe delivery of products without the need for direct physical contact between the robot and the recipient.
[5]"Delivery Robots in Logistics: A Review of Recent Advances and Challenges" by	2021	Nalinaksh Vyas and Arindam Ghosh.	Challenges in Adoption and Deployment: The authors discuss the challenges associated with the adoption and deployment of delivery robots in logistics. This includes regulatory hurdles, safety concerns, public acceptance, and the need for standardized guidelines and policies.
[6] "Delivery Robots for Last-mile Logistics	2019	Vikas Kumar, Luciana Moreira, and Johan Scholliers.	Last-Mile Delivery Challenges: The paper highlights the challenges associated with last-mile delivery in logistics

Operations: A Review”			operations and discusses how delivery robots can address these challenges. It addresses issues such as increasing urbanization, traffic congestion, high delivery costs, and environmental concerns.
[7]"Design and Development of a Low-cost Autonomous Delivery Robot for Indoor Environments"	2018	Fernando Sancho, Andrés Olivares, and Luis Payá	Low-Cost Autonomous Delivery Robot: The paper focuses on the design and development of a low-cost autonomous delivery robot specifically designed for indoor environments. The authors discuss the physical design of the robot, including its size, structure, and components. Navigation and Localization in Indoor Environments: The authors address the navigation and localization challenges encountered by the autonomous delivery robot in indoor environments.
[8]"A Review on the Use of	2019	Nanda Kumar D, Hanumanthappa	Applications and Benefits of Autonomous

<p>Autonomous Unmanned Aerial Vehicles for Delivery Services" by</p>		<p>M, and Vijaya Kumar B P.</p>	<p>Unmanned Aerial Vehicles (UAVs) for Delivery Services: The paper reviews the applications and benefits of autonomous unmanned aerial vehicles (UAVs) in the context of delivery services. It discusses how UAVs can be used for various delivery tasks, including medical supplies, food delivery, e-commerce packages, and emergency services.</p>
--	--	---------------------------------	--

1.2 Challenges and Problems

Outdoor navigation for robot systems presents several challenges that need to be overcome for the robots to navigate autonomously in outdoor environments. Some of the significant challenges are:

- **Complex Terrain:** Outdoor environments are diverse, with uneven surfaces, varying slopes, and terrains that can be challenging for robot navigation.
- **Uncertainty and Variability:** The environment is continually changing, with unpredictable weather conditions, unpredictable obstacles like rocks, debris, and other environmental factors that can impact the robot's performance.
- **Sensor Limitations:** Outdoor environments can present challenges for sensors, such as GPS signal disruption, low visibility due to environmental conditions, and sensor failure due to harsh weather conditions.
- **Power Constraints:** Outdoor robots typically require a more substantial power source than indoor robots, and finding a reliable power source can be challenging.
- **Communication Challenges:** Communication in outdoor environments is complicated, with limited communication range and signal interference due to various factors such as terrain, weather conditions, and obstacles.
- **Safety Concerns:** Safety is a critical concern in outdoor robot navigation, and the robot must be designed to operate safely in the presence of pedestrians, other vehicles, and animals.
- **Legal and Regulatory Hurdles:** There are regulatory and legal requirements that must be met for outdoor robot systems to operate, such as permits, safety requirements, and liability issues.
- **Overcoming these challenges** requires careful planning, advanced technology, and a thorough understanding of the environment in which the robot will operate

1.3 Objectives

- To use state of art GPS and IMU for improving the sensor accuracy and limitation
- To improve security concerns raised during delivery with the means of secure QR code based magnetic locker
- Autonomy infrastructure using only depth cameras, eliminating high cost LIDAR's. And using different forms of localization would help the robot understand its environment.

1.4 Methodology

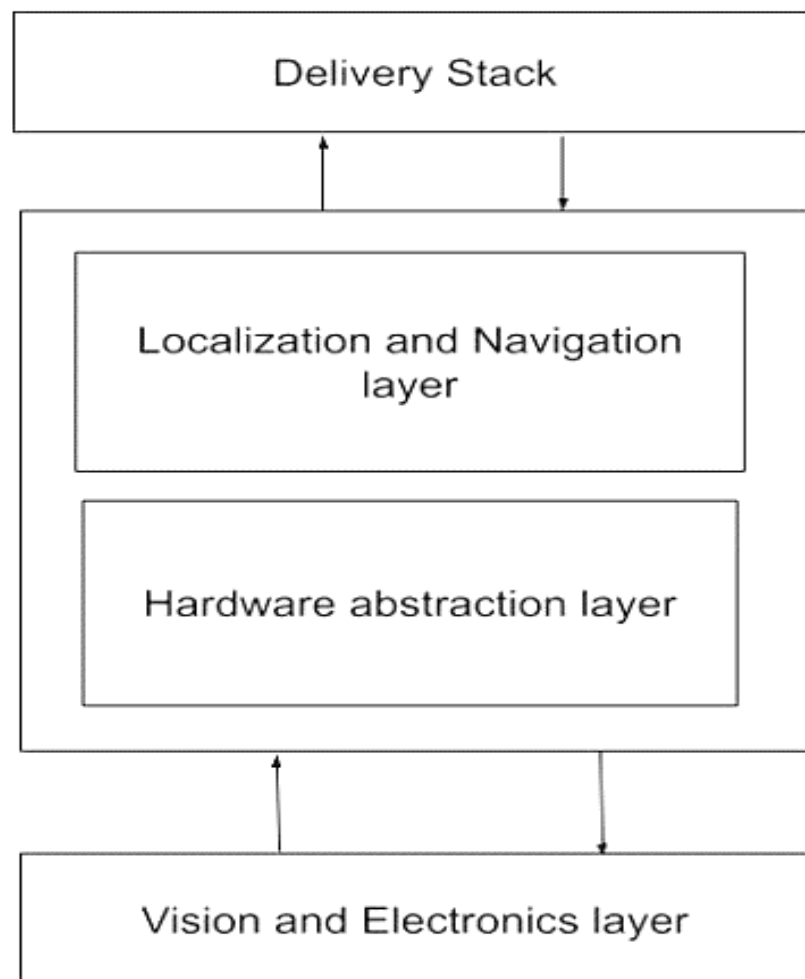


Figure 1.4.1 Block diagram 1

The Ditto robot utilizes two main layers to communicate with its sensors and the base station: the Hardware Abstraction Layer and the Vision and Electronics Layer.

1. Hardware Abstraction Layer: This layer serves as an intermediary between the robot's hardware components and the higher-level software. It provides a standardized interface for accessing and controlling the hardware resources. The Hardware Abstraction Layer enables efficient communication and control of various components within the robot.

2. Vision and Electronics Layer: This layer forms the first layer of the Ditto robot's hardware. It comprises several key components, including:

- **Motors and Motor Driver:** The motors are responsible for the movement of the robot in free space. They receive commands from the higher-level software, such as speed and direction, and convert them into physical motion. The motor driver facilitates the control and operation of the motors by regulating power and signals.

- **Battery:** The battery is an essential component that provides power to the robot. It supplies the necessary electrical energy to operate the motors, electronics, and other components of Ditto.

- **Camera:** The camera is a crucial part of the Ditto robot's vision system. It captures visual information from the robot's surroundings, allowing it to perceive and understand its environment. The camera data can be processed by the higher-level software to enable various functionalities, such as object recognition, navigation, and mapping.

The Vision and Electronics Layer integrates these components to enable the Ditto robot to perceive its environment through vision, control its movement using motors, and sustain its operation through the battery. This layer forms the foundation for higher-level functionalities and applications that utilize the sensor data and interact with the robot's hardware.

The vision system of the Ditto robot is vital for road segmentation, object detection, and QR code detection, enabling effective navigation. Here's an explanation of each functionality:

- 1. Road Segmentation:** Road segmentation involves distinguishing the road or ground plane from other areas in the camera's field of view. This capability allows the robot to identify safe areas for navigation. The camera captures visual information, and advanced image processing techniques are applied to identify and segment the road region accurately.
- 2. Object Detection:** Object detection involves identifying and locating specific objects or obstacles in the robot's environment. The Ditto robot's camera captures visual data, and sophisticated algorithms are used to detect objects. This involves analyzing the camera feed using techniques like convolutional neural networks (CNNs) to accurately identify and locate objects of interest. Object detection enables the robot to perceive its surroundings, navigate safely, and avoid obstacles.
- 3. QR Code Detection:** QR code detection refers to the ability to recognize and interpret QR codes present in the camera's field of view. QR codes are two-dimensional barcodes containing various types of information, such as URLs or text. The camera captures visual data, and software algorithms are employed to identify and decode QR codes. This functionality allows the robot to scan QR codes on signage or labels for navigation instructions or to access specific information related to the environment.

By utilizing these vision-based functionalities, the Ditto robot can navigate its surroundings effectively. It can accurately segment the road area, detect objects to avoid collisions, and recognize QR codes to acquire relevant information or instructions. The camera's visual input, processed by sophisticated software algorithms, empowers the robot to make informed decisions regarding its navigation path and interact with its environment intelligently.

Next comes the localization layer, traditionally GPS are used to navigate and give the robot understanding of its surroundings to localize. GPS (Global Positioning System) plays a significant role in localization, which involves determining the precise position or coordinates of an object or entity. Here's how GPS contributes to localization:

- 1. Satellite-based Positioning:** GPS relies on a network of satellites orbiting the Earth. These satellites continuously transmit signals that are received by GPS receivers. By receiving signals from multiple satellites, a GPS receiver can calculate its position based on the time it takes for the signals to reach it.

The GPS receiver uses the time differences between the signals to triangulate its position accurately.
- 2. Trilateration:** Trilateration is a key concept in GPS localization. By receiving signals from at least three satellites, a GPS receiver can determine its position in three-dimensional space. Each satellite acts as a reference point, and the distances between the receiver and the satellites are calculated based on the time delay of the signals. Using the known positions of the satellites, the GPS receiver can calculate its own position through trilateration algorithms.
- 3. Geographic Coordinate System:** GPS provides location information in the form of geographic coordinates, typically latitude and longitude. These coordinates define a precise position on the Earth's surface. By utilizing GPS and receiving signals from multiple satellites, a GPS receiver can accurately determine its latitude and longitude coordinates. These coordinates enable localization by specifying the exact position of an object or entity on the Earth.
- 4. Navigation and Mapping:** GPS localization is widely used for navigation and mapping purposes. GPS receivers can continuously track their positions and provide real-time updates, enabling users to determine their current location accurately. GPS is employed in various applications, such as vehicle navigation systems, mobile mapping, and geolocation services. By combining GPS with mapping data, users can navigate from one location to another with precise directions and track their movements in real-time.

In summary, GPS plays a crucial role in localization by utilizing satellite-based positioning, trilateration, and providing geographic coordinates. GPS enables accurate positioning and is widely used for navigation, mapping, and various other applications requiring precise localization information.

But in ditto we are not using GPS, instead we will be using QR code-based localization since we know the map of the environment we are deploying on and also, we know the route taken. Level 2 autonomy provides assisted autonomy, that is the operator will be assisting the robot navigate in case of any trouble risen during delivery.

Level 2 autonomy in localization:

Level 2 autonomy in robots, including delivery robots, primarily focuses on vehicle control and assistance systems rather than robot localization. Robot localization refers to the ability of a robot to determine its own position within its environment. However, we can provide insights on how certain technologies associated with level 2 autonomy can indirectly contribute to robot localization:

1. Sensor Integration: Level 2 autonomy often involves the integration of various sensors, such as cameras, LiDAR, and radar, into the robot's design. These sensors provide data about the robot's surroundings, which can be used for localization purposes. For example, LiDAR sensors can generate a 3D map of the environment, allowing the robot to localize itself relative to known landmarks or objects in its vicinity.

2. Perception and Mapping: The sensors employed in level 2 autonomy help in perception and mapping of the robot's environment. This information can be used to create a map of the surroundings, including obstacles, landmarks, or reference points that aid in localization. By continuously updating and referencing this map, the robot can determine its position and navigate accordingly.

3. Simultaneous Localization and Mapping (SLAM): SLAM is a technique used to simultaneously build a map of the environment and determine the robot's position within that map. Although SLAM is typically associated with higher levels of autonomy, some aspects of SLAM, such as mapping, can be incorporated into level 2 autonomy systems. By utilizing sensor data and algorithms, the robot can update its map and estimate its position relative to that map, facilitating localization.

It's important to note that while level 2 autonomy systems provide sensor data that can assist with robot localization, they may not directly address the complete localization process. Localization often requires more advanced algorithms and techniques, such as probabilistic approaches (e.g., particle filters), sensor fusion, or use of additional sensors like odometry or GPS. Therefore, while level 2 autonomy systems contribute to the perception and understanding of the robot's environment, further localization algorithms and techniques may be necessary to achieve precise and accurate localization within the robot's operating environment. Hence with the help of QR codes placed at every milestone helps the operator see the logs and also the robot know how far the destination is from its location.

Delivery layer comprises of all these layers adding and supporting each other. Detail explanation is given in chapter 4



Fig 1.4.2 Delivery robot designed by AI 1



Fig 1.4.3 Starship robot fleet

1.5 Hardware and Software tools used.

Ditto uses following hardware and software specifications, these tools are responsible for the control and monitoring of the robot during operation.

Hardware:

1. Raspberry Pi 4B+ with Ubuntu server 20.04
2. Temperature sensor (DHT11)
3. Ultrasonic or sonars
4. OAK-D Lite depth camera
5. Locking system

Software:

1. Open street maps
2. ROS or Robot Operating System (Noetic)
3. Open VINO (AI models) along with depth

More information about the above mentioned tools are provided in chapter 3

CHAPTER 2

2. Basic Theory behind working of Ditto

The outdoor navigation of mobile robots using only depth cameras involves using depth data to enable a robot to autonomously navigate through an outdoor environment. Depth cameras are capable of capturing depth information, which can be used to create a 3D representation of the environment. By analyzing this information, a mobile robot can determine its location and plan a path to its destination.

The basic theory of outdoor navigation using depth cameras involves three main stages: mapping, localization, and path planning.

The mapping stage involves creating a map of the environment using the depth data captured by the depth cameras. The robot moves through the environment while capturing depth data, and this data is processed to create a map of the environment. The map includes information about obstacles, landmarks, and other relevant features that the robot can use to navigate.

The mapping stage usually entails gathering information about the environment using sensors like cameras, LiDAR, and GPS. The information is then combined and processed to produce an environment plan that the robot can use to navigate. Depending on the particular application, localization can be done offline or in real-time. Due to the presence of obstacles like trees, buildings, and uneven terrain, mapping can be especially difficult in outdoor settings.

Mapping in robotics involves gathering information about the environment using sensors like cameras, LiDAR, and GPS. These sensors capture data that is processed and combined to create an environment plan, enabling the robot to navigate effectively. However, mapping in outdoor settings presents unique challenges. Obstacles such as trees, buildings, and uneven terrain can hinder sensor data collection and accuracy. Lighting conditions, including bright sunlight or shadows, can also impact sensor performance. Additionally, outdoor environments are subject to changes over time, requiring the mapping system to adapt and update the environment representation accordingly. Furthermore, GPS limitations in urban canyons or dense foliage areas can affect the accuracy of positioning information. To overcome these challenges, robust sensor technologies are employed, such as combining camera images with LiDAR data for comprehensive environment perception. Advanced data processing techniques like sensor fusion and feature extraction are utilized to create accurate environment representations. Mapping algorithms, including SLAM, estimate the

robot's position and simultaneously construct a map of the surroundings. Additionally, intelligent strategies are implemented to handle environmental changes, ensuring the map stays up to date. Overall, outdoor mapping in robotics requires a combination of reliable sensors, sophisticated data processing algorithms, and adaptive strategies to overcome obstacles and create an accurate understanding of the environment.

The robot can plan its route and use it to navigate once an environment map has been made. The localization stage involves determining the robot's current location within the mapped environment. This is done by comparing the depth data captured by the depth camera with the map previously created in the mapping stage. The robot's position is estimated by matching the depth data with the known features on the map.

The path planning stage involves generating a path from the robot's current location to its destination. The robot uses the map and its current location to determine the best path to take, taking into account obstacles and other relevant features.

The planning step entails creating a plan using the environment map and other pertinent data, such as the robot's current condition and the task or goal it must complete. The strategy outlines a series of steps the robot should take to complete the task or goal, taking into consideration any limitations like short battery life or environmental obstacles.

For autonomous robots, a variety of planning algorithms can be used, from straightforward ones like A* search to more intricate ones like probabilistic roadmaps and quickly investigating random trees. (RRTs). The particular job, environment, and robot constraints all influence the algorithm that is chosen.

The depth camera technology used in outdoor robot navigation has some limitations. Depth cameras can be affected by environmental factors such as lighting conditions, weather, and the reflective properties of the surfaces they are detecting. These factors can cause inaccuracies in the depth data, which can result in errors in the robot's navigation.

Depth camera technology, while useful for outdoor robot navigation, does have some limitations that can impact its performance.

Here are further details on the factors that can affect depth cameras in outdoor environments:

1. Lighting conditions: Depth cameras rely on emitting and sensing infrared light to measure distances. However, outdoor lighting conditions can vary significantly, including bright sunlight, shadows, or low-light environments. Direct sunlight can interfere with the camera's infrared emitter, making it difficult to accurately measure depth. Shadows can also affect the camera's ability to capture depth information, leading to inaccuracies in the perceived distances.

2. Weather conditions: Adverse weather conditions such as rain, fog, or snow can have a negative impact on depth camera performance. Droplets of water or particles in the air can scatter or absorb the emitted infrared light, reducing its effectiveness in measuring depth. Similarly, snowflakes or raindrops can obstruct the camera's view and cause distortions in the captured depth data.

3. Reflective surfaces: The reflective properties of surfaces in the environment can affect the accuracy of depth measurements. Highly reflective surfaces, such as glass, mirrors, or shiny metal, can reflect the infrared light emitted by the depth camera, leading to erroneous depth readings. This can cause objects to appear closer or farther than they actually are, potentially affecting the robot's navigation decisions.

To mitigate these limitations, several approaches can be adopted:

- Calibration and synchronization: Depth cameras can be calibrated and synchronized with other sensors to compensate for lighting variations and improve accuracy. For example, combining depth data with visual information from cameras can help overcome limitations caused by reflective surfaces.

- Filtering and post-processing: Depth data can be filtered and processed using algorithms to reduce noise and enhance accuracy. Techniques like outlier removal, temporal filtering, and surface reconstruction can improve the quality of depth measurements and provide a more reliable representation of the environment.

- Sensor redundancy: By integrating multiple sensors, including depth cameras, LiDAR, or radar, the robot can benefit from the strengths of each sensor while compensating for their individual limitations. Sensor fusion techniques can be employed to combine data from different sources and generate a more robust perception of the environment.

Overall, while depth cameras can face challenges in outdoor environments due to lighting conditions, weather, and reflective surfaces, careful calibration, filtering techniques, and sensor fusion can help overcome these limitations and enhance the accuracy of depth data for outdoor robot navigation.

Despite these limitations, the use of depth cameras for outdoor robot navigation has significant advantages. It provides a cost-effective and efficient solution for robot navigation in outdoor environments, as it eliminates the need for additional sensors such as GPS or LiDAR. It also enables robots to navigate in areas where GPS signals may be disrupted, such as urban canyons or heavily forested areas.

In conclusion, the basic theory of outdoor navigation of mobile robots using only depth cameras involves mapping the environment, localizing the robot's position within the map, and generating a path from the robot's current location to its destination. Although depth camera technology has some limitations, it provides a cost-effective and efficient solution for robot navigation in outdoor environments. Here we use QR code base localization for the robot to understand how far it is from the delivery destination, it also gives the operator a log of operation. It also uses QR code-based authentication for fail proof mechanism of the robot locking stack .

CHAPTER 3

3. Tool Description

3.1 Hardware Components

3.1.1 Raspberry Pi 4B+:

The Raspberry Pi 4 Model B is the latest product in the Raspberry Pi range, boasting a 64-bit quad core processor running at 1.5GHz, dual-band 2.4GHz and 5GHz wireless LAN, Bluetooth 5.0/BLE, true Gigabit Ethernet, and PoE capability via a separate PoE HAT.

- A high-performance 64-bit quad-core processor
- Dual display support with resolutions up to 4K via a pair of micro-HDMI ports
- Hardware video decoding up to 4Kp60
- 4 GB of RAM
- A connection to the dual-band wireless local area network 2.4/5.0 GHz
- Bluetooth 5.0 / Gigabit Ethernet / USB 3.0 / PoE features (via a separate HAT PoE add-on module)



Fig 1: Raspberry Pi 4B+

3.1.2 Temperature Sensor (DHT11):

DHT11 sensor consists of a capacitive humidity sensing element and a thermistor for sensing temperature. The humidity sensing capacitor has two electrodes with a moisture holding substrate as a dielectric between them. Change in the capacitance value occurs with the change in humidity levels. The IC measure, process this changed resistance values and change them into digital form. The temperature range of DHT11 is from 0 to 50 degree Celsius with a 2-degree accuracy. Humidity range of this sensor is from 20 to 80% with 5% accuracy. The sampling rate of this sensor is 1Hz .i.e. it gives one reading for every second. DHT11 is small in size with operating voltage from 3 to 5 volts. The maximum current used while measuring is 2.5mA.

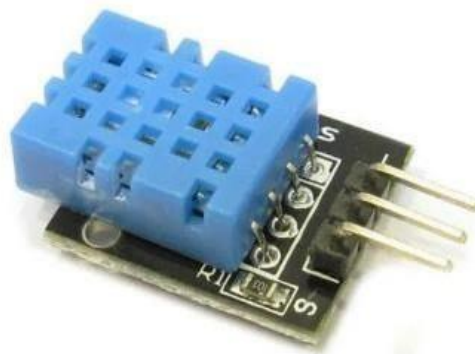


Fig 4: DHT11

3.1.3 Ultrasonic or Sonar:

An ultrasonic sensor is an electronic device that measures the distance of a target object by emitting ultrasonic sound waves, and converts the reflected sound into an electrical signal. Ultrasonic waves travel faster than the speed of audible sound (i.e. the sound that humans can hear). Ultrasonic sensors have two main components: the transmitter (which emits the sound using piezoelectric crystals) and the receiver.

In order to calculate the distance between the sensor and the object, the sensor measures the time it takes between the emission of the sound by the transmitter to its contact with the receiver. The formula for this calculation is $D = \frac{1}{2} T \times C$ (where D is the distance, T is the time, and C is the speed of sound ~ 343 meters/second). For example, if a scientist set up an ultrasonic sensor aimed at a box and it took 0.025 seconds for the sound to bounce back, the distance between the ultrasonic sensor and the box would be:

$$D = 0.5 \times 0.025 \times 343$$



Fig 3.1.3 Ultrasonic sensor 1

3.1.5 OAK-D Lite depth camera:

OAK-D-Lite is smaller, lighter and uses less power; compared to OAK-D. The main differences are:

Dimensions and Weight

- Mono cameras have lower resolution (640x480 instead of 1280x800)
- Initial versions of OAK-D-Lites (KickStarter batch) had no IMU. Current batches have BMI270.
- There is no power jack, as most users just used the USB-C for power delivery, which provides 900mA at 5V and is enough for most use-cases. However, some functions (e.g. inference, video encoding) can lead to large current spikes, so there is a chance hosts like RPi won't be able to provide enough power. In that case, you should use Y-adapter.
- Width: 91 mm
- Height: 28 mm
- Length: 17.5 mm
- Baseline: 75 mm
- Weight: 61 g

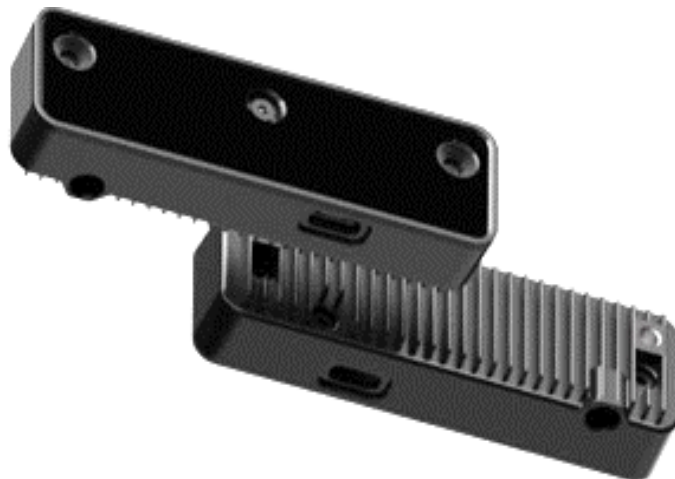


Fig 3.1.5 OAK-D lite

CHAPTER 4

4. IMPLEMENTATION

4.1 Hardware design

4.1.1 Overview

The robot is intended to function as an autonomous mobile robot platform for various purposes such as transportation in industrial sectors, food and medicine delivery in hospitals, and unmanned missions.

As well as for security. Our goal is to create an autonomous delivery mobile robot that can autonomously deliver a package from A to B on the SETJU campus.

4.1.2 Design Criteria

There were major constraints in hardware design. These are outlined below:

1. **Modularity:** In order to easily add units or parts to the robot, the robot platform has to be modular. These units or parts may be additional navigation sensors, room for payload, extra on-board power, or various devices for effective human-machine interface.
2. **Low-cost Production:** Even though mobile robots are available in the market, they tend to be expensive, thus increasing research and development costs. Further, the use of a ready-made robot will increase the cost of production even more in case of mass volume production.
3. **Truncated Construction:** In the prototyping phase, the construction is kept simple and truncated in order to use minimal resources and to focus on designated functionality.
4. **Suitability of Environmental Conditions:** Robot is planned to be used in outdoor environments, which means that the robot has to move on the roads and be able to pass over small obstacles. Additionally, electronic equipment on the robot must be protected.
5. **Originality :** To contribute to scientific research and development, the robot has to be different and new. The hardware and software structure of the robot has been designed by taking into consideration the above criteria

4.1.3 Hardware structure

The robot was first designed using fusion 360 software, the visualization and measurements were taken according to the robot expected structure. The robot was then checked for its stability using stress management tools available on fusion 360. Later, for fabrication we used wood since they are sustainable and can last long and light weight. The robot's body was fabricated according to the design but availability of motors and wheels changed our design idea to more modular and light weight model.

The robot currently consists of two layers.

1. Vision and Electronics layer :

This layer consists of all the sensors mentioned above, which will help us communicate with the robot and the robot with its environment.

2. Locking system layer :

This layer is made of magnetic lock based trigger system, which gets triggered only when the customer authenticates his/her order. The locking system layer as a small compartment for storing food items and delivering it to customers at required location.



Fig 4.1.3 CAD model

The robot is designed according to the design criteria and further field tests have been started. The environmental considerations are fully met and a robust structure has been developed. The robot weighs about 5kg and has a payload capacity of 2kg

4.2 Building the prototype

Building a prototype using wood and MDF (Medium-Density Fiberboard) sheet materials offers a cost-effective and versatile approach for testing and refining product designs.

Wood provides structural integrity and durability, while MDF sheets offer smooth surfaces for precision work. In this two-page answer, we will explore the steps involved in building a prototype using wood and MDF sheets, highlighting the materials, tools, and techniques utilized.

Material Selection: Wood provides strength, stability, and natural aesthetics, making it suitable for structural elements of the prototype. Common types of wood used include pine, plywood, or hardwood, depending on the desired characteristics and load requirements. MDF sheets, on the other hand, offer a smooth and uniform surface, ideal for creating precise and dimensionally accurate parts.

Planning and Design: Before starting the prototype construction, it is essential to have a well-defined design plan. This includes detailed sketches, technical drawings, and dimensions. The design plan helps in visualizing the final product, determining the required materials and measurements, and ensuring accuracy during the construction process. Here we used the CAD design which was created for ditto.

Initially we faced challenges in identifying the material thickness and size required. We built a bigger prototype but was not feasible so we had to revise the prototype to smaller version

Joinery and Assembly: To assemble the prototype, various joinery techniques can be employed. Common methods include butt joints, miter joints, dowel joints, or using fasteners like screws or nails. The choice of joinery depends on the design requirements, load-bearing capacity, and the desired aesthetics of the prototype.



Fig 4.2.1 Pre body of Robot



Fig 4.2.2 Finished Model of Robot

4.3 Softwares used

4.1.4 FUSION 360



Fig 10: CAD Model of Robot

Fusion 360 is a powerful computer-aided design (CAD), computer-aided manufacturing (CAM), and computer-aided engineering (CAE) software developed by Autodesk. It offers a comprehensive suite of tools and features that enable designers and engineers to create, simulate, and manufacture complex 3D models and prototypes.

With Fusion 360, users can design 3D models using parametric modeling techniques, which allow for easy modification and iteration of designs. The software provides a wide range of tools for creating geometric shapes, assemblies, and parts, along with advanced features like surface modeling and sculpting. It also supports collaborative design by enabling multiple users to work on the same project simultaneously, facilitating real-time collaboration and feedback.

In addition to its robust design capabilities, Fusion 360 incorporates CAM functionality, allowing users to generate toolpaths and G-code for CNC (Computer Numerical Control) machining. This integration of CAD and CAM within a single platform streamlines the manufacturing process, as users can seamlessly transition from design to production. Fusion 360 also includes simulation and analysis tools, enabling engineers to validate designs, perform stress and thermal analysis, and optimize their models for performance and functionality.

One of the key advantages of Fusion 360 is its cloud-based nature, which enables users to access their designs and project data from anywhere with an internet connection. This cloud-based approach allows for easy collaboration, data management, and version control. Additionally, Fusion 360 offers a vast library of tutorials, forums, and learning resources,

making it accessible to users of all skill levels and providing ample support for mastering the software.

In summary, Fusion 360 is a comprehensive CAD/CAM/CAE software that empowers designers and engineers to create, simulate, and manufacture 3D models and prototypes. With its robust design tools, integrated CAM capabilities, simulation features, and cloud-based collaboration, Fusion 360 provides a complete solution for the entire product development process.

Ditto was first created using FUSION 360 to design and analyse the prototype strength and weakness. The latter was used to generate URDF or unified robot description format to simulate the results in ROS and GAZEBO.

4.1.5 GAZEBO

Gazebo, is a widely used open-source software tool that integrates with the Robot Operating System (ROS) to simulate and visualize robotic systems in a virtual environment. It provides a highly flexible and powerful platform for developing and testing robot algorithms, as well as for simulating complex scenarios and environments. Gazebo offers a range of features that make it an essential tool for researchers, developers, and robotics enthusiasts.

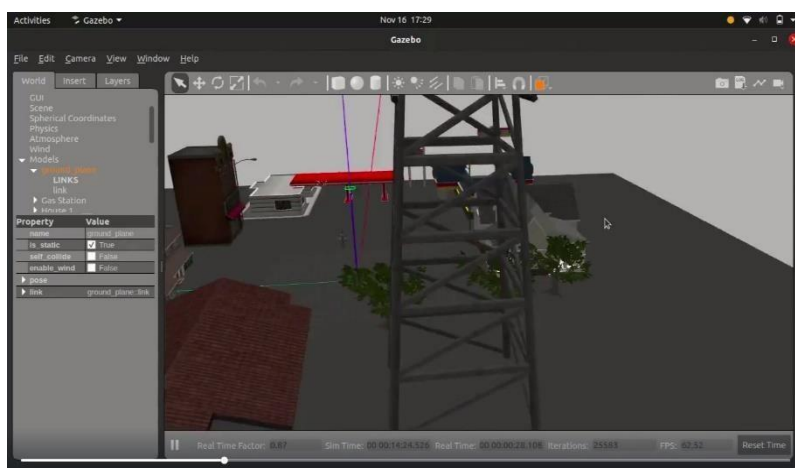


Fig 4.1 Gazebo simulated model

One of the key strengths of Gazebo is its realistic physics engine, which accurately models the dynamics of objects and robots within the simulated environment. This allows users to test and refine their control algorithms and system designs in a virtual setting before deploying them on physical robots. Gazebo supports various types of sensors, including cameras, LIDAR, and depth sensors, enabling the simulation of perception systems and the

generation of sensor data for algorithm development and testing.

Gazebo is designed to seamlessly integrate with ROS, the de facto standard middleware for robotics development. This integration enables users to leverage the vast ecosystem of ROS packages and libraries, allowing for easy transfer of code and data between simulation and real-

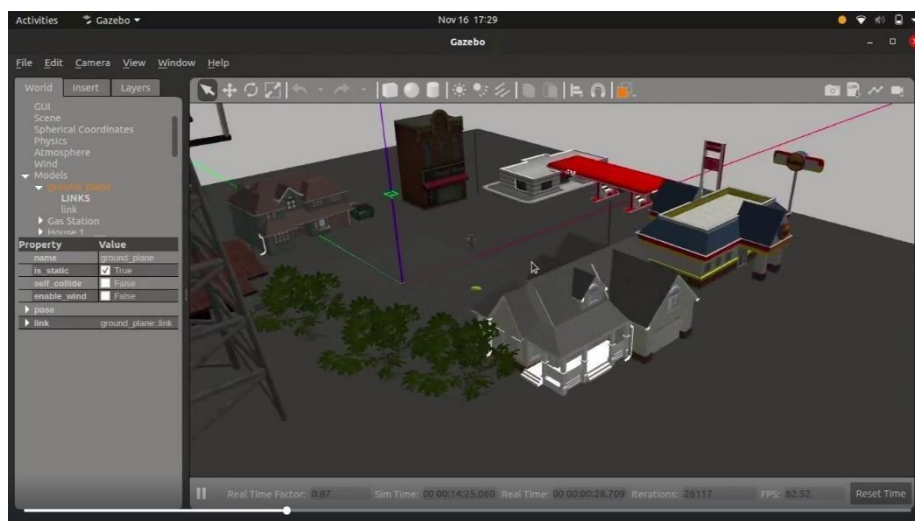


Fig 4.3 Gazebo simulated model

world robots. Gazebo also provides ROS interfaces that enable real-time control and visualization of simulated robots, making it an excellent tool for developing and testing robot applications within the ROS framework.

Furthermore, Gazebo supports the creation and customization of virtual worlds and environments. Users can design and simulate various types of terrains, buildings, and objects to replicate real-world scenarios or create entirely new environments for testing and experimentation. This flexibility, combined with its robust physics engine and ROS integration, makes Gazebo an indispensable tool for prototyping, algorithm development, and system validation in the field of robotics.

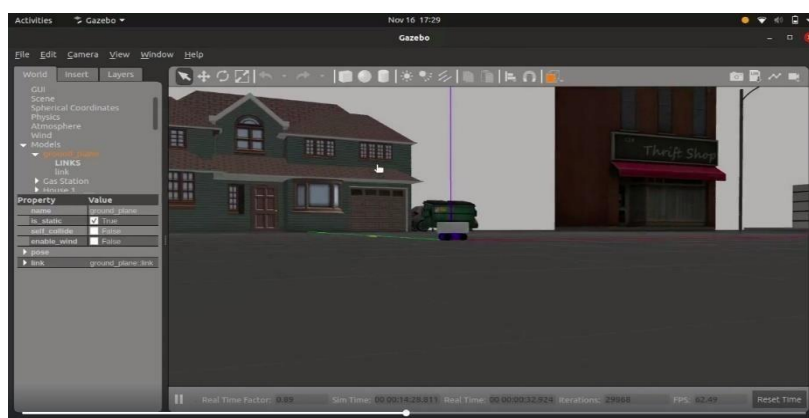


Fig 4.3 Gazebo simulated model
www.ijert.org

4.1.6 ROS



ROS (Robot Operating System) is a flexible and powerful framework widely used in robotics for developing and controlling robotic systems. It provides a collection of libraries, tools, and conventions that facilitate communication, coordination, and integration of various components within a robotic system. ROS offers several key features that make it a popular choice for researchers, engineers, and hobbyists in the robotics community.

One of the primary strengths of ROS is its modular architecture, which enables the development of complex robotic systems by breaking them down into smaller, reusable components called nodes. These nodes can be independently developed, tested, and integrated into a larger system, fostering code reusability and system scalability. ROS also provides a robust communication infrastructure, allowing nodes to exchange messages, share data, and coordinate their activities through a publish-subscribe model or service calls.

Another significant advantage of ROS is its extensive library of pre-built software packages, known as ROS packages. These packages cover a wide range of functionalities, including perception, control, planning, and simulation. By leveraging these packages, developers can save time and effort by utilizing existing code and algorithms, accelerating the development process. The ROS package ecosystem is continually growing, with contributions from a large and active community, making it a valuable resource for both beginners and experienced users.

Additionally, ROS offers powerful tools for visualization, debugging, and analysis of robotic systems. Tools such as RViz enable real-time visualization of sensor data, robot models, and algorithms, aiding in the development and debugging of perception and control systems. ROS also supports simulation environments like Gazebo, allowing users to test and validate their algorithms and system designs in virtual environments before deploying them on physical robots.

4.2 Localization and Navigation

Before we get in to the localization layer, let's talk about the Vision and electronics layer. Vision and electronics layer consist of battery to power the motors and our processor. The robot uses a 12V 9amp battery ,the robot is 4WD which uses differential drive mechanism to maneuver the robot indifficult situation.

The differential drive mechanism is a type of vehicle steering and propulsion system commonly used in robotics and wheeled vehicles. It consists of two separately driven wheels on either side of the vehicle, where each wheel can be independently controlled in terms of speed and direction. By adjusting the relative speeds and directions of the wheels, the vehicle can move forward, backward, turn in place, or follow curved paths.

In a differential drive system, turning is achieved by driving the wheels on opposite sides of the vehicle at different speeds or in opposite directions. When the wheels rotate at equal speeds in the same direction, the vehicle moves straight. To turn, the wheel on one side rotates faster or in the opposite direction compared to the other wheel. This speed differential creates a torque that causes the vehicle to turn.

Skid steering, also known as tank steering or zero-radius turning, is another mechanism used in certain vehicles. In skid steering, instead of using a differential, the wheels on either side are independently driven and controlled. To turn, the wheels on one side rotate in the opposite direction or stop while the other side continues to move. This differential wheel speed creates a skidding motion, causing the vehicle to turn around a pivot point.

Ackerman steering mechanism, on the other hand, is commonly used in vehicles with multiple wheels, such as cars. It is a mechanism that enables the wheels to turn at different angles to facilitate smoother turning. In Ackerman steering, the inside wheel of the turning radius is steered at a sharper angle than the outside wheel. This configuration helps to minimize tire scrubbing and provides better control and stability during turns. In summary, the differential drive mechanism is characterized by independently driven wheels on either side of the vehicle, where turning is achieved by varying the speed or direction of the wheels. Skid steering involves independently controlling the wheels on each side to create a skidding motion for turning. Ackerman steering is a mechanism used

in vehicles with multiple wheels, where the inside and outside wheels are steered at different angles during turns for improved maneuverability and stability.

Ditto uses the differential drive mechanism which improves the steering capabilities and maneuvering tasks easy. Vision layer consist of depth camera which is used for depth estimation, road segmentation and object detection. These AI techniques are mainly used by the robot to perceive its environment.

Depth estimation using depth cameras is a process of determining the distance to objects in a scene from the camera's viewpoint. Depth cameras, also known as depth sensors or depth perception cameras, capture depth information in addition to color or RGB data. These cameras utilize various technologies such as time-of-flight, structured light, or stereo vision to measure the depth of the scene.

Time-of-flight (ToF) cameras emit light or infrared signals and measure the time it takes for the signal to bounce back from objects in the scene. By calculating the round-trip time, the camera can estimate the distance to the objects. ToF cameras provide real-time depth measurements and are commonly used in applications such as gesture recognition and robotics.

Structured light cameras project a pattern of light onto the scene and analyze the deformation of the pattern on the objects to calculate depth. By comparing the known pattern with the deformed pattern captured by the camera, the system can infer the depth information. Structured light cameras are often used in 3D scanning, object recognition, and augmented reality applications.

Stereo vision cameras use two or more cameras placed at different positions to capture the scene from slightly different viewpoints. By analyzing the disparities between corresponding pixels in the images captured by the cameras, the system can calculate the depth of objects. Stereo vision cameras provide accurate depth estimation and are commonly used in robotics, autonomous vehicles, and 3D reconstruction.

Once the depth data is obtained, various algorithms and techniques can be applied to process and interpret the depth information. These may include depth image filtering, point cloud generation, or surface reconstruction. Depth estimation can enable applications such as object recognition, scene understanding, augmented reality, virtual reality, robotics navigation.

In summary, depth estimation using depth cameras involves capturing depth information from a scene using technologies like time-of-flight, structured light, or stereo vision.

This depth data allows for the calculation of distances to objects and enables a wide range of applications in various fields.

Road segmentation:

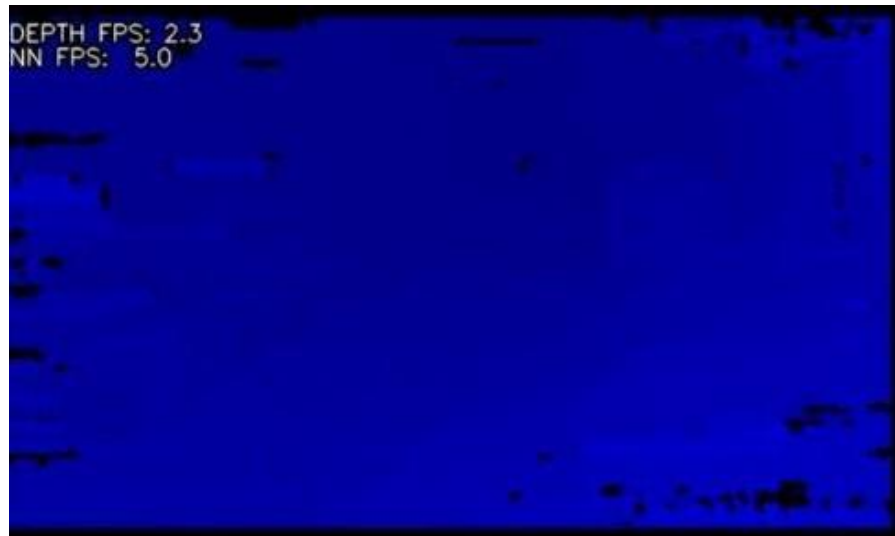
Road segmentation is a computer vision task that involves identifying and segmenting the regions of an image or video frame that correspond to the road or the driving surface. It is an essential step in many applications such as autonomous driving, road monitoring, and traffic analysis. Road segmentation algorithms aim to accurately distinguish between the road and non-road areas, which may include sidewalks, buildings, vehicles, or vegetation.



1. Color-based segmentation: This method utilizes the color information of the road surface to distinguish it from the background. Since roads often exhibit specific color characteristics (e.g., shades of gray or brown), color thresholding or clustering algorithms can be employed to separate the road pixels from the rest.



2. Texture-based segmentation: Road surfaces often possess distinctive textures that differ from other objects in the scene. Texture analysis methods, such as Gabor filters, can be applied to capture these unique textural features and segment the road based on texture similarity.



3. Semantic segmentation: This approach involves training deep learning models, such as convolutional neural networks (CNNs), on large annotated datasets to classify each pixel in the image as road or non-road. These models leverage the hierarchical features learned from the training data to achieve accurate pixel-level segmentation.

4. LiDAR-based segmentation: In scenarios where LiDAR sensors are available, 3D point cloud data can be utilized for road segmentation. LiDAR sensors measure the distance to objects, allowing for the extraction of ground points and subsequent road segmentation based on elevation or geometric properties.

It's important to note that road segmentation can be challenging due to variations in lighting conditions, weather conditions, occlusions, or complex urban environments. Therefore, a combination of multiple techniques or the use of more advanced robustness of road segmentation systems. Successful road segmentation can provide valuable information for various downstream tasks in autonomous driving, such as lane detection, obstacle detection, and path planning, contributing to safer and more efficient transportation systems.

Here in Ditto, we use semantic segmentation for road segmenting. Semantic segmentation is an approach to road segmentation that involves training deep learning models, particularly convolutional neural networks (CNNs), to classify each pixel in an image as either road or non-road. Semantic segmentation aims to understand the content of an image at a pixel level and assign a semantic label to each pixel.

1.Dataset preparation: Semantic segmentation models require a large dataset of annotated images where each pixel is labeled with its corresponding class (e.g., road, sidewalk, building, etc.). The dataset should include diverse road scenes to capture different lighting conditions, road types, and environmental factors. Manual annotation or labeling tools are used to create ground truth labels for each pixel in the training images.

2.Model architecture: Convolutional neural networks (CNNs) are commonly used for semantic segmentation tasks. These networks consist of multiple layers of convolutional, pooling, and up sampling operations. The architecture may include encoder-decoder structures, skip connections, or atrous convolutions to capture both local and global information. Popular CNN architectures for semantic segmentation include U-Net, Fully Convolutional Networks (FCN), and DeepLab.

3.Training process: During the training phase, the labeled dataset is used to train the CNN model. The input images are fed into the network, and the model gradually learns to predict the road or non-road class for each pixel. The training process involves optimizing the model's parameters by minimizing a loss function that measures the difference between predicted labels and ground truth labels. Common loss functions used in semantic segmentation include cross-entropy loss and dice loss.

4.Inference and prediction: Once the model is trained, it can be used for road segmentation on new, unseen images. The model processes an input image and generates a prediction map where each pixel is assigned a class label. The road class label represents the segmented road regions in the image. Post-processing techniques like smoothing, morphological operations, or conditional random fields (CRF) can be applied to refine the segmentation results.

5. Evaluation and refinement: The performance of the semantic segmentation model is assessed using evaluation metrics such as pixel accuracy, intersection over union (IoU), or mean average precision (mAP). If the results are not satisfactory, the model can be refined by adjusting hyperparameters, increasing the training dataset size, or fine-tuning the architecture.

Semantic segmentation based on deep learning techniques has shown promising results in road segmentation tasks. It can handle complex scenes, handle variations in lighting conditions, and learn intricate features to distinguish road regions from non-road regions. However, it requires a significant amount of labeled training data and computational resources for training and inference.

We are using road segmentation model which is already given by depth AI, this road

Segmentation data is lightweight and readily available, there is a faster computation of this model on raspberry pi.

- **Object detection:** Object detection is a computer vision task that involves identifying and localizing objects of interest within an image or a video. It aims to determine not only the presence of objects but also their locations and extent. Object detection is widely used in various applications such as autonomous driving, surveillance, robotics, and image understanding.

Here's a high-level explanation of the object detection process:

1. Input and preprocessing: Object detection algorithms take an input image or a video frame as their input. Prior to detection, preprocessing steps such as resizing, normalization, or color space conversions may be performed to prepare the input data for analysis.

2. Feature extraction: Object detection algorithms analyze the input data to extract relevant features that represent objects. Traditional methods use handcrafted features like Histogram of Oriented Gradients (HOG) or Haar-like features. However, deep learning-based approaches, particularly convolutional neural networks (CNNs), have become prevalent due to their ability to learn hierarchical and discriminative features directly from data.

3. **Region proposal:** In many object detection algorithms, a region proposal step is employed to generate potential regions or bounding boxes in which objects may be present. These proposals define the regions to be further analyzed for object classification and localization. Techniques like Selective Search, Edge Boxes, or Region Proposal Networks (RPNs) are commonly used for generating these region proposals.

4. **Object classification:** In this step, the regions generated from the previous step are reclassified into different object categories. The extracted features or the region proposals are passed through a classifier, typically a CNN, which predicts the presence and the class labels of the objects within each region. Classification can be performed using soft ax or sigmoid activation functions to estimate the probabilities or confidence scores for each object class.

Localization: Along with object classification, object detection algorithms aim to accurately localize the objects within the image. This involves estimating the bounding boxes that tightly enclose the objects. Localization can be achieved by regressing the coordinates of the bounding boxes based on the region proposals or directly predicting the coordinates as part of the detection network.

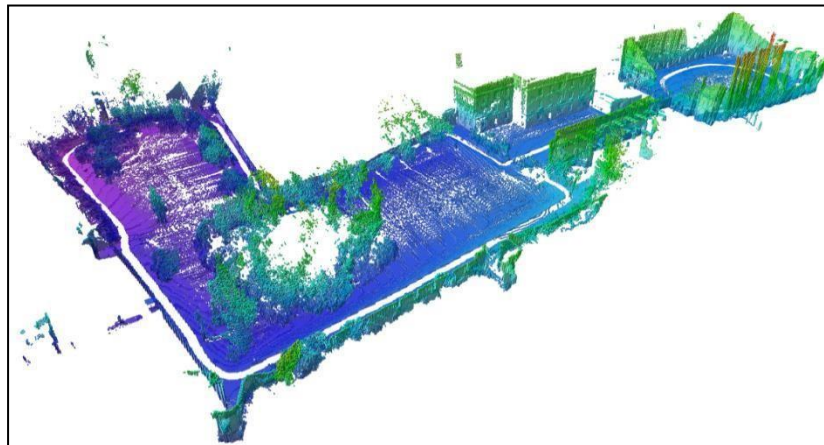


Fig: Lidar based localization

Post-processing: After classification and localization, post-processing steps are typically applied to refine the detection results. Non-maximum suppression (NMS) is a commonly used technique to remove redundant bounding boxes and retain the most confident and non-overlapping detections. Thresholding on the confidence scores may also be applied to filter out detections below a certain confidence threshold.

Object detection algorithms can vary in their complexity and performance depending on the specific approach used, the size of the object categories, the dataset used for training, and the computational resources available. Recently, advanced object detection architectures like Faster R-CNN, YOLO (You Only Look Once), and SSD (Single Shot MultiBox Detector) have gained popularity due to their improved accuracy and efficiency.

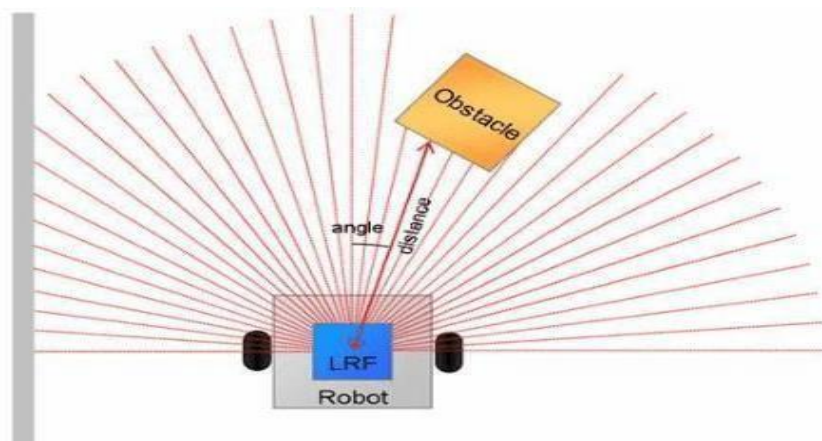
Object detection plays a crucial role in numerous applications, enabling machines to perceive and understand their surroundings, perform automated actions, and make informed decisions based on the detected objects. Ditto uses YOLOv4 Model YOLOv4 (You Only Look Once version 4) is an advanced object detection algorithm known for its improved accuracy and speed. It builds upon previous versions (YOLOv3, YOLOv2, YOLO) and incorporates several enhancements to achieve better performance. One of the key features of YOLOv4 is its backbone network, which utilizes the CSPDarknet53 architecture. This modified version of Darknet improves the network's ability to capture rich and multi-scale features from the input image. Additionally, YOLOv4 introduces the PANet (Path Aggregation Network) module as its neck network, enabling the fusion of features at different scales. This helps the model handle objects of various sizes more effectively.

The detection head of YOLOv4 consists of multiple layers responsible for predicting bounding boxes, object classes, and confidence scores. The architecture of the detection head is designed to capture object details with different aspect ratios and scales, improving localization and classification accuracy. YOLOv4 incorporates advanced training techniques, such as data augmentation methods (mosaic augmentation and random shape training) and training strategies like Cosine Annealing Scheduler and IoU-aware training. These techniques enhance the model's robustness and convergence during training.

Furthermore, YOLOv4 focuses on model size and speed optimizations without compromising accuracy. Techniques like Mish activation, CIOU loss, and Weighted Residual Connections are employed to improve performance and efficiency. It is worth noting that YOLOv4 is an open-source framework, and specific implementation details may vary.

1. Obstacle detection and avoidance: Delivery robots need to navigate through complex and

dynamic environments while avoiding obstacles such as pedestrians, vehicles, or other objects. Object detection algorithms enable the robot to identify and localize these obstacles in real-time, allowing it to plan safe and collision-free paths. By detecting and avoiding obstacles, delivery robots can ensure the successful and efficient completion of their delivery tasks.



2. Package detection and handling: Delivery robots need to accurately detect and locate packages for pickup and delivery. Object detection algorithms enable the robot to identify and localize packages, enabling it to approach, grasp, and transport them securely. This ensures that the robot can effectively interact with the packages it needs to handle during its delivery operations



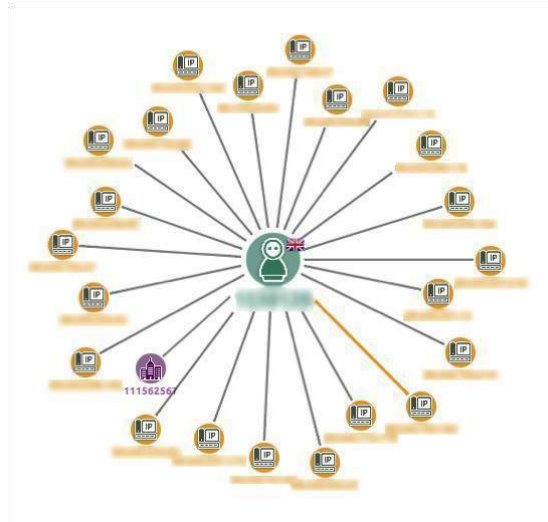
3. Delivery point identification: Object detection can also be used to identify specific delivery points, such as mailboxes, drop-off locations, or designated areas for package delivery. By recognizing and localizing these specific points, the delivery robot can precisely navigate and position itself for package drop-off or pickup. This helps ensure accurate and efficient delivery operations.



4. Traffic sign and signal recognition: Delivery robots operating in urban or shared spaces may encounter various traffic signs and signals, such as stop signs, crosswalks, or traffic lights. Object detection algorithms can be employed to recognize and interpret these traffic signs and signals, enabling the robot to comply with traffic rules and ensure safe navigation through intersections or pedestrian-heavy areas.



5. Security and anomaly detection: Object detection algorithms can be used to identify suspicious or unauthorized objects in the vicinity of the delivery robot. This enhances the robot's security capabilities, allowing it to identify potential threats or anomalies and take appropriate actions, such as notifying operators or avoiding the area.



By leveraging object detection technology, delivery robots can navigate their environment safely, handle packages efficiently, identify delivery points accurately, and enhance their overall functionality and performance. The ability to detect and interact with objects in their surroundings enables delivery robots to operate autonomously and effectively carry out their delivery tasks.

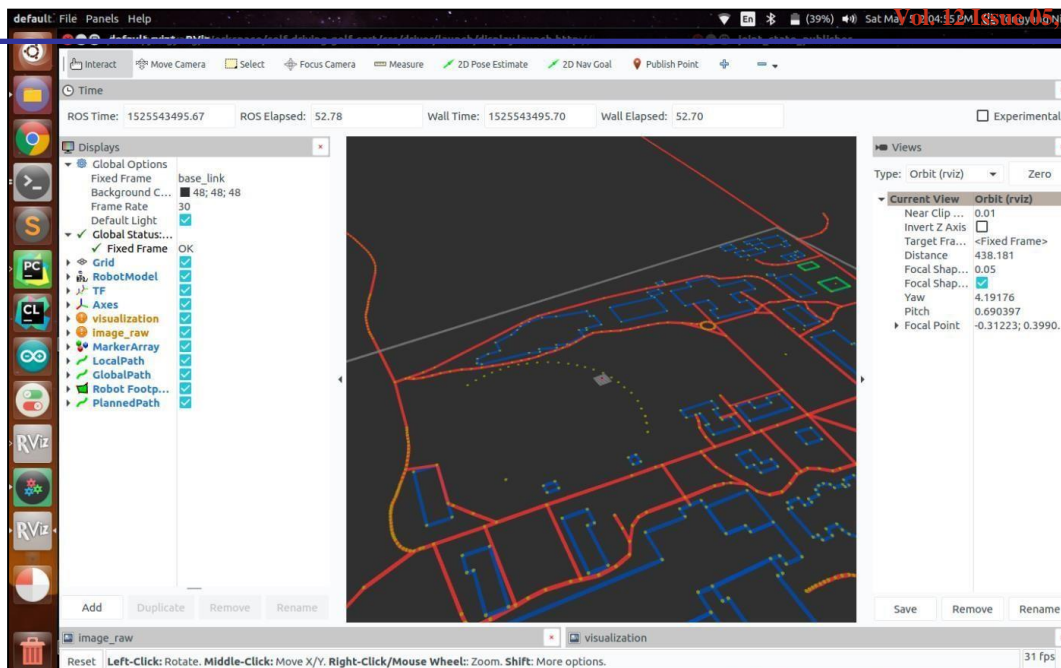
4.4.2 Localization

Localization refers to the process of determining the position and orientation of a robot or an object in its environment. In the context of delivery robots, localization is crucial for accurate navigation and efficient delivery operations.

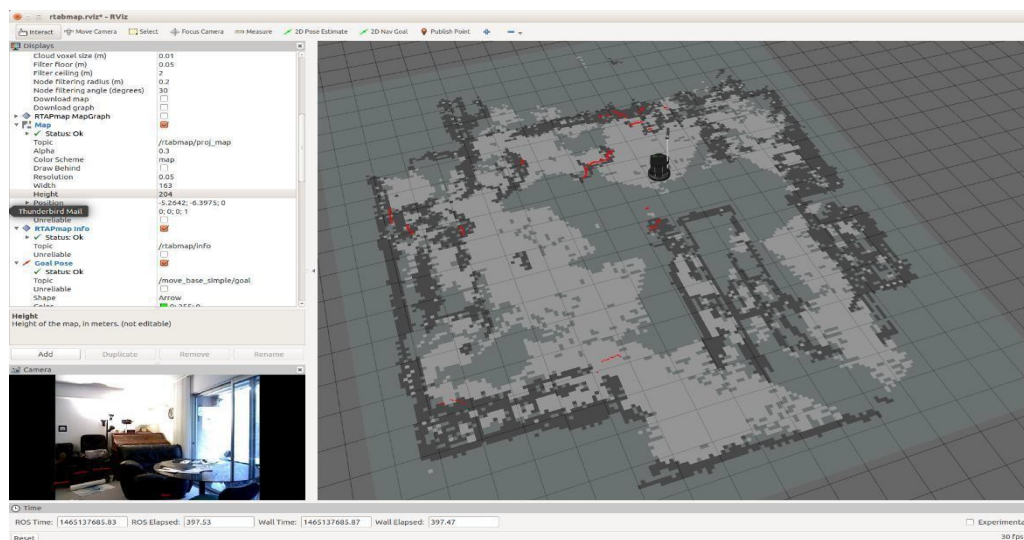
Here are different localization techniques commonly used for delivery robots:

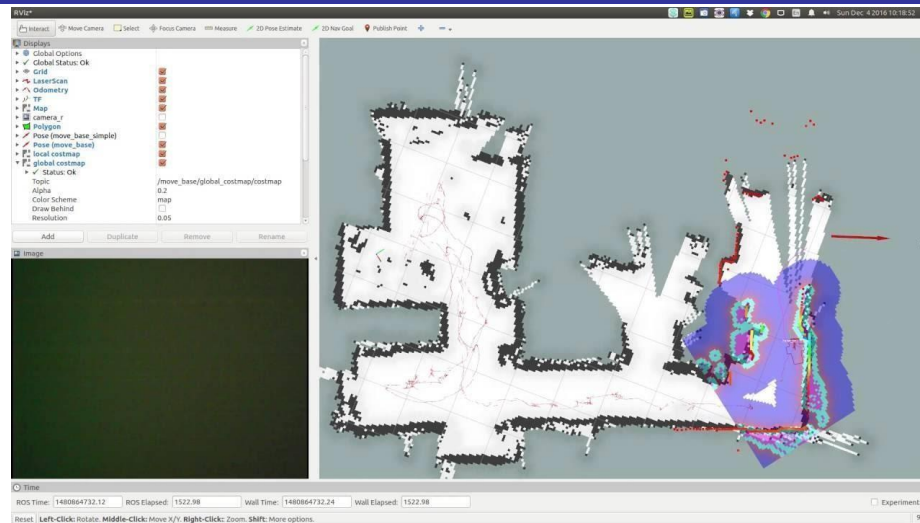
1. **GPS-based Localization:** Global Positioning System (GPS) is a widely used localization technique that relies on satellite signals to determine the robot's position on the Earth's surface. GPS provides a global coordinate system, allowing robots to obtain their latitude, longitude, and sometimes altitude information.

However, GPS signals can be affected by obstacles, signal loss, or multi-path interference, which can lead to inaccuracies in urban or indoor environments. To improve accuracy, GPS can be combined with other localization techniques.



- 2. Visual Localization:** Visual localization involves using cameras or visual sensors to extract visual information from the environment and match it with pre-existing maps or landmarks. This technique can utilize techniques like visual odometry, feature matching, or simultaneous localization and mapping (SLAM) to estimate the robot's position based on visual cues. Visual localization is advantageous as it can work in various environments, both indoors and outdoors, and doesn't rely on external infrastructure. However, it may be sensitive to lighting conditions, occlusions, or dynamic changes in the environment.





3. **LiDAR-based Localization:** Light Detection and Ranging (LiDAR) sensors emit laser beams to measure the distance to surrounding objects. By scanning the environment, creating a point cloud, and comparing it with a known map, LiDAR-based localization can estimate the robot's position and orientation. LiDAR provides accurate 3D spatial information and is commonly used in combination with other sensors for robust localization.
4. **Radio-based Localization:** Radio-based techniques, such as radio frequency identification (RFID) or Bluetooth beacons, can be used for localization. RFID tags or beacons placed in the environment emit signals that the robot can detect and use to estimate its position. This technique is particularly useful for localization in indoor environments with predefined beacons or tags.
5. **Sensor Fusion and Particle Filters:** Sensor fusion combines data from multiple sensors, such as cameras, LiDAR, odometry, or inertial measurement units (IMUs), to improve localization accuracy. Particle filters or probabilistic localization algorithms are often used to fuse sensor data and estimate the robot's position and orientation based on probabilistic models.

These are some of the localization techniques commonly employed in delivery robots. The

choice of technique depends on factors such as the environment, accuracy requirements, available infrastructure, and the robot's specific capabilities. Many modern localization approaches combine multiple techniques to achieve robust and accurate position estimation for delivery robots.

Here in Ditto we are using visual based localization, visual markers such as QR codes. The QR codes displayed will be scanned by the robot and will have a rough estimate on how far it is from the destination. This method can only be used when the map and route is known to the robot operator.

4.4.3 Navigation

Navigation in delivery robots refers to the process by which these robots plan and execute their movements while delivering packages or goods. It involves perceiving the environment, making decisions, and physically moving through the environment.

Delivery robots are equipped with sensors such as cameras, LiDAR, ultrasonic sensors, and proximity sensors to perceive their surroundings. They create a map of the environment using simultaneous localization and mapping (SLAM) techniques. This map helps them understand the layout and plan their paths.

Path planning algorithms consider the robot's current location, destination, obstacles, traffic rules, and other constraints to determine the best path. Obstacle avoidance is crucial for safe navigation, and robots detect obstacles using their sensors, adjust their paths, or employ reactive control to avoid collisions.

Localization techniques like GPS, odometry, and visual odometry help robots estimate their positions accurately. They use actuators, such as wheels or legs, to physically move through the environment. The control system translates navigation commands into appropriate movements.

Delivery robots may need to communicate with a central control system or other robots for coordination. Communication systems enable smooth operation and collaboration.

Navigation in delivery robots involves software algorithms, hardware components, and real-time decision-making to transport goods efficiently and safely. The specific techniques and technologies vary depending on the robot's design, environment, and delivery tasks.

Here in ditto, we are not depended on costly Lidars and GPS module. Instead we are using the above mentioned vision layer along with all the necessary AI models which will be

running parallelly to help the robot perceive its environment.

Along with that we are also adding DHT11 sensors to get the temperature for our packages that would be delivered.

QR code based assistive navigation system:

The robot will receive its delivery point using our webapp, this webapp mainly is used to get the delivery and pickup location and then send the otp qrcode to the customer. Once

That is done, the robot starts its tasks, and the QR code used will tell us the distance of the robot from the pick up to destination and give the operator a set of logs which he will be able to understand. Since the use of GPS is not present, this type of QR code localization can be used.

Disadvantages of using this form of QR based localization is:

- We should know the exact path taken and map of the location
- Proper localization is not observed
- The odometry of the robot is not known other than the wheel encoders
- The robot may or may not be able to detect QR code depending on various Lighting conditions.
- Navigation can be improved with the usage of GPS sensors.

The robot since it is in level 2 autonomy, the operator will be able to operate the bot, and assist the robot in case of emergency situation.

4.4 Delivery Stack

Delivery stack of ditto consists of all layers combined. The vision and electronics layer uses the navigation and localization layer with hardware abstraction, to reach the goal of the robot. Ditto once reached its destination will deliver the package as intended. Ditto uses a rack and pinion mechanism for controlling the movement of the lock.

The rack and pinion mechanism is a mechanical arrangement commonly used for converting rotary motion into linear motion. It consists of two main components: a rack and a pinion gear.

The rack is a flat or cylindrical bar with teeth along its length. These teeth mesh with the teeth of the pinion gear, which is a small cylindrical gear with a shaft at its center. The pinion gear rotates, causing the rack to move in a linear direction.

As the pinion gear turns, its teeth engage with the teeth of the rack, pushing or pulling it in the desired direction. This interaction between the teeth of the rack and pinion allows the rotational

motion of the pinion to be translated into linear motion along the rack.

The rack and pinion mechanism finds extensive use in various applications, including steering systems in automobiles, machine tools, robotics, and other machinery that require precise linear motion control

In summary, the rack and pinion mechanism is a mechanical arrangement comprising a rack with teeth and a pinion gear. The rotational motion of the pinion gear engages with the teeth of the rack, converting it into linear motion. This mechanism is commonly employed in applications where precise linear motion is required, such as steering systems and machine tools.

So when the QR code which was sent by the web app while placing the order, will be used to authenticate the order by using the QR based detection algorithm.

The QR code detection algorithm

```
import cv2

import numpy as np
# Load the input image

image = cv2.imread("input_image.jpg")
# Convert the image to grayscale

gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
# Create a QR code detector object

qr_detector = cv2.QRCodeDetector()
# Detect and decode QR codes in the image

data, points, _ = qr_detector.detectAndDecodeMulti(gray)
# Check if any QR codes are
detectedif len(data) > 0:
print("QR Code detected!")
# Iterate over each detected QR code
for i in range(len(data)):

    # Extract the bounding box coordinates of the QR code
    x, y, w, h = points[i].reshape(-1)
    # Draw a rectangle around the QR code cv2.rectangle(image,
(x, y), (x + w, y + h), (0, 255, 0), 2)
    # Print the decoded data

print("QR Code {}: {}".format(i+1, data[i]))
```

```
# Display the image with detected QR codes
```

```
cv2.imshow("QR Code Detection", image)
```

```
cv2.waitKey(0)
```

```
else:
```

```
    print("No QR Code detected.")
```

This above pseudo code is used along with ROS image capture to scan the QR code and trigger the locking stack. It is important to secure the package since there might be chances of public interference and also for the below mentioned causes:

Ensuring package security is of utmost importance for a delivery robot due to several reasons. First, it helps prevent theft of valuable items, such as packages, groceries, or sensitive documents, during transportation, especially in public spaces or unmonitored areas. Second, it safeguards packages from tampering or damage, ensuring that fragile or perishable contents remain intact. By maintaining the security of packages, the robot also upholds the integrity of their contents and prevents unauthorized access or interference.

Moreover, package security is crucial for maintaining customer trust. Customers expect their items to be delivered securely and without any disruptions. By prioritizing package security, delivery robots contribute to building positive customer relationships and ensuring satisfaction. Additionally, compliance with legal regulations and liability considerations necessitate package security. Certain items, such as confidential documents, medical supplies, or hazardous materials, may have specific security requirements. Adhering to these measures helps the delivery robot and the service provider

Furthermore, package security extends beyond physical protection. Delivery robots may handle sensitive data related to the delivery process, such as customer addresses or payment information. It is essential to implement robust cybersecurity measures to safeguard this data from unauthorized access or breaches, thus preserving the privacy and confidentiality of customer information.

In summary, ensuring package security is crucial for delivery robots to prevent theft, protect contents from tampering or damage, maintain customer trust, comply with legal requirements, and safeguard sensitive data. Addressing these security aspects allows delivery robots to perform their tasks responsibly and reliably.

Chapter 5: Results and Discussion

The evaluation of Ditto, a level 2 autonomous delivery bot, showcased promising results. Ditto effectively utilizes a QR code-based localization technique to precisely navigate its surroundings. This technique allows the bot to determine its position accurately and efficiently, enabling seamless movement during delivery operations. Moreover, Ditto's implementation of a depth camera significantly enhances its road segmentation and object detection capabilities. The depth camera enables the bot to discern road boundaries with high accuracy, ensuring smooth and reliable navigation in various environments. Additionally, the advanced object detection functionality empowers Ditto to identify obstacles along its path, thereby minimizing the risk of collisions and ensuring safe delivery operations.

A standout feature of Ditto is its QR code-based locking mechanism, which offers a fail-proof solution for secure item transportation. The locking system utilizes a rack and pinion mechanism, which enhances the reliability and robustness of the mechanism. This design ensures that the locking stack remains securely closed until the designated QR code is scanned, providing a high level of protection for the delivered items. The implementation of this mechanism strengthens Ditto's ability to fulfill its delivery objectives effectively and efficiently, instilling confidence in both the users and recipients of the bot's services. Overall, the evaluation results highlight the significant potential of Ditto as a reliable and autonomous delivery bot, demonstrating its effectiveness in navigating diverse environments, detecting obstacles, and ensuring secure item transportation through its innovative QR code-based locking system.



Fig 5.1 Ditto at SETJU

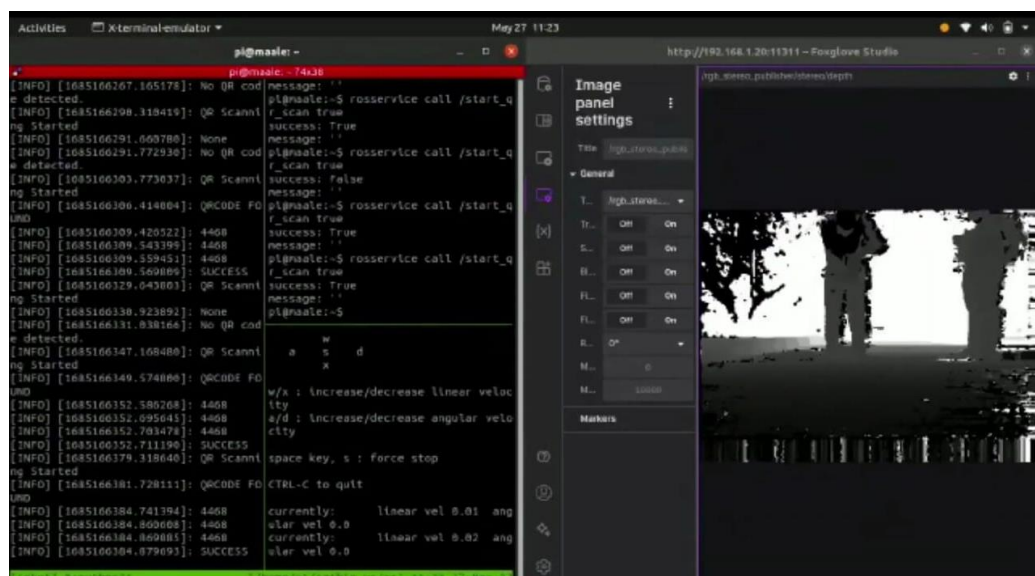


Fig 5.2 Ditto's camera perspective

Conclusion and future scope

Ditto, an autonomous delivery robot operating at level 2, showcases the immense potential in the realm of package transportation. Its self-navigation capabilities and efficient operations offer a promising solution for the future of delivery services. However, the development and implementation of delivery robots like Ditto also open up avenues for further advancements and enhancements in this field.

In conclusion, Ditto serves as a prime example of how delivery robots can revolutionize the transportation of packages. With its autonomy and robust safety features, it ensures secure and efficient delivery operations, leading to increased productivity and customer satisfaction.

Looking forward, the future scope for delivery robots, including Ditto, is vast. Ongoing research and development endeavors are focused on enhancing their navigation abilities to seamlessly adapt to complex urban environments. This involves refining obstacle avoidance mechanisms, precise localization techniques, and optimizing path planning algorithms to improve reliability and efficiency.

Additionally, the integration of robotics and Internet of Things (IoT) technologies provides opportunities for advanced package tracking and monitoring. Real-time updates on delivery status, temperature control for perishable items, and secure package handling mechanisms can be integrated into future iterations of delivery robots.

Moreover, collaboration between delivery robots and existing transportation networks holds potential for optimized last-mile delivery solutions. By coordinating with traditional delivery vehicles or leveraging public transportation systems, efficiency can be improved while reducing road congestion. In summary, Ditto represents a significant advancement in autonomous delivery robots, highlighting their potential to transform package transportation. Continued research and development efforts will refine their capabilities, enabling seamless navigation, improved safety, and enhanced environmental interaction. The future of delivery robots holds great promise, with advancements in AI, IoT integration, and collaboration with existing transportation infrastructure shaping the future of the logistics industry.

REFERENCES

- [1] "An Autonomous Vehicle for Parcel Delivery in Urban Areas" Alexander Buchegger, Konstantin Lassnig, Stefan Loigge, Clemens Muhlbacher, and Gerald Steinbauer, 2018 21st International Conference on Intelligent Transportation Systems (ITSC)Maui,Hawaii, USA.
- [2] "Design and Development of Autonomous Delivery Robot" Aniket Gujarathi, Akshay Kulkarni, Unmesh Patil, Yogesh Phalak, Rajeshree Deotalu, Aman Jain, Navid Panchi under the guidance of Dr. Ashwin Dhabale and Dr. Shital S. Chiddarwar-2020
- [3] "Object Recognition of Outdoor Environment by Segmented Regions for Robot Navigation" Dae-Nyeon Kim, Hoang-Hon Trinh, and Kang-Hyun Jo, Graduate School of ElectricalEngineering, University of Ulsan-2019.
- [4] "An Autonomous Delivery Robot to Prevent the Spread of Coronavirus in Product Delivery System" Murad Mehrab Abrar, Raian Islam, Md. Abid Hasan Shanto, Ahsanullah University of Science and Technology-2021.
- [5] "Delivery Robots in Logistics: A Review of Recent Advances and Challenges" by Nalinaksh Vyas and Arindam Ghosh. (2021)
- [6]"Design and Development of a Low-cost Autonomous Delivery Robot for Indoor Environments" by Fernando Sancho, Andrés Olivares, and Luis Payá. (2018)
- [7] "Delivery Robots for Last-mile Logistics Operations: A Review" by Vikas Kumar, Luciana Moreira, and Johan Scholliers. (2019)
- [8] "A Comprehensive Study on the Application of Drones in Logistics and Supply Chain Management" by Kunpeng Zhang, Yuan Li, and Jiang Zhang. (2020).
- [9]"A Review on the Use of Autonomous Unmanned Aerial Vehicles for Delivery Services" by Nanda Kumar D, Hanumanthappa M, and Vijaya Kumar B P. (2019)
- [10] Website: Starship Technologies - <https://www.starship.xyz/> YouTube: Starship Autonomous Delivery Robots - https://www.youtube.com/watch?v=RcP_e7nNTgk
- [11] Website: KiwiBot - <https://www.kiwibot.com/>
YouTube: KiwiBot Food Delivery - <https://www.youtube.com/watch?v=ZvQgEaCTsN4>
- [12] Website: Nuro - <https://nuro.ai/>
YouTube: Nuro Self-Driving Delivery Vehicle - <https://www.youtube.com/watch?v=nPmjO5Z9l4A>

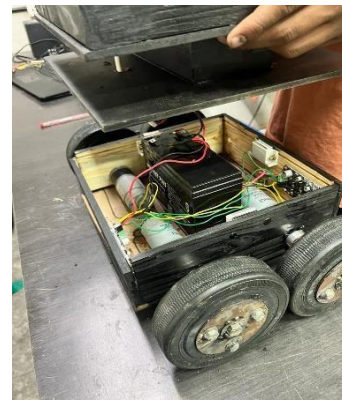
[13] Website: Robby Technologies - <https://roby.io/>

YouTube: Robby Delivery Robot - <https://www.youtube.com/watch?v=bIL12zHINRI>





[14] Website: Marble - <https://www.marble.io/>

YouTube: Marble Delivery Robots - <https://www.youtube.com/watch?v=X6XtpStAT8U>

APPENDIX-I PHOTOGRAPHS



Information regarding students

Name	Email ID	Phone Number	Address	Landline Number	Place ment details	Photograph
Dennis Mcleaord	Dennis171101@gmail.com	9160200871	Flat No:506,JRR Towers,Srinagar 7/4,Guntur,Andhra Pradesh.522002 Placement: Not placed	nil	Unplaced	
Athish Anand Kumar	athishanandkumar@gmail.com	9880617668	#002 Ciro apartment 1st main 1st cross behind HDFC Bank opposite to Guru Darshan hotel Vijaya bank layout Bannerghatta road Bangalore 560076	nil	Unplaced	
Haarvish chevula	haarvish@gmail.com	8341735406	13-238-2 mungamuri vari street ,beside oxford school,Nellore,Andhrapradesh 524001 Placement not placed trying offcampus	nil	unplaced	
Siddarth D	Siddarth.dayasagar@gmail.com	8197566989	#22/38,'padmaka mala',near shiva apts, nagarbhavi,bengaluru-72	nil	Unplaced	

BATCH PHOTOGRAPH ALONG WITH GUIDE

