CPE 470/670 Final Report
Fall 2023

Keaton Clark
Undergrad

Andrew DeBolt
Undergrad

Mahdi Shasavar
Graduate

Colin McAlvain
Graduate

Pamella Nipay
Undergrad

***Abstract*—The final project for the fall 2023 semester requires that a Pioneer robot placed at a unknown location inside a known environment be able to autonomously navigate outside of the environment to unknown location labeled with an Aruco Marker. Our design incorporated use of the camera and the Aruco libaries to find the opening in the environment and the marker at the final location as well as the encoders to precisely move inside the environment. Our design completed the task in just over a 60 seconds.**

I. Introduction

The final project presents a challenging yet intriguing scenario in which our Pioneer robot is placed within an enclosed environment, denoted as room ABCD, and is equipped with the objective of self-navigating to a given ArUco target. The environment is characterized by walls and openings, and the robot is aided by ArUco tags and sensors for localization and navigation. The goal is to successfully guide the robot out of room ABCD through the designated opening, drive it to a specific ArUco tag, and ultimately align it with the target ArUco tag. This task requires a blend of sensor-based navigation, decision-making, and precise control to achieve success.

In this report, we will detail our approach to solving this autonomous navigation challenge. We will discuss the strategies employed, the sensors utilized, and the challenges faced during the process. Our aim is to complete the task in the shortest time or by implementing multiple sensors for enhanced performance.

II. Components

- Pioneer Robot
  - Definition: A Pioneer Robot is a type of mobile robot platform commonly used in robotics research and education. It is designed for various applications, including autonomous navigation, mapping, and research purposes.
  - Description: Pioneer Robots are equipped with sensors and actuators that enable them to move in their environment, collect data, and perform tasks autonomously or under human control. They serve as a versatile platform for developing and testing robotic algorithms and applications.
- Raspberry Pi V.3
  - Definition: Raspberry Pi V.3 refers to the third version of the Raspberry Pi single-board computer. It is a compact, affordable, and low-power computing device used for a wide range of applications, including robotics, Internet of Things (IoT), and do it yourself (DIY) projects.
  - Description: Raspberry Pi V.3 features a quad-core ARM processor, HDMI output, USB ports, GPIO pins, and wireless connectivity options. It provides a cost-effective solution for running various software and programming languages, making it a popular choice for building embedded systems and robotics projects.
- Lidar
  - Definition: Lidar, short for ”Light Detection and Ranging,” is a remote sensing technology that uses laser light to measure distances and create detailed 3D maps of objects and environments. It is commonly used in robotics for perception and navigation.
  - Description: Lidar sensors emit laser pulses and measure the time it takes for the light to bounce off objects and return. This data is used to create high-resolution 3D maps, detect obstacles, and enable robots to navigate safely and accurately in complex surroundings.
- Encoder
  - Definition: An encoder is a device used in robotics and automation to measure the position, speed, and direction of a rotating shaft or wheel. It converts mechanical motion into electrical signals.
  - Description: Encoders are essential for precise control in robotics. They provide feedback to the robot's control system, allowing it to determine the current position and adjust as needed. Encoders are commonly used in wheel and motor control systems.
- Logitech Camera
  - Definition: A camera is a device that captures visual information by recording images or videos. Cameras are used in robotics to provide images and video streams of their surroundings or objects of interest.
  - Description: Cameras are essential components in robotics and computer vision applications, such as object recognition, navigation, and perception. They

capture high-quality images or video streams, allowing robots to perceive and interact with their surroundings. They provide valuable visual input for robots to make informed decisions and carry out tasks effectively.

- ArUco Markers
  - Definition: ArUco markers are 2D barcodes with a unique pattern used for marker-based augmented reality (AR) and computer vision applications. They are often used in robotics for object tracking and localization.
  - Description: ArUco markers are easy to detect and identify in images or video streams. They serve as reference points for robots to determine their position and orientation relative to the markers. This makes ArUco markers valuable for robot localization and navigation tasks.
- Robot Operating Systems (ROS)
  - Definition: ROS, or Robotic Operating Systems, is an open-source framework for developing and controlling robotic systems. It provides a set of libraries and tools for tasks such as hardware abstraction, communication, and robot behavior development.
  - Description: ROS enables developers to create modular and reusable software components for robots. It simplifies the development of complex robotic applications by providing a standardized way for different software modules to communicate and interact.
- Python
  - Definition: Python is a high-level, versatile programming language known for its simplicity and readability. It is widely used in robotics and automation for developing software to control robots and perform various tasks.
  - Description: Python's ease of use and extensive libraries make it a popular choice for robot programming. It allows developers to quickly prototype and implement algorithms for tasks such as navigation, image processing, and machine learning, making it a valuable tool in the field of robotics.
- Roscore
  - Definition: Roscore is a fundamental component of ROS. It acts as a master node responsible for managing the ROS computation graph, including topics, services, and parameter servers.
  - Description: Roscore provides a centralized point for communication and coordination among different nodes in a robotic application. It initializes the ROS system and ensures that nodes can discover and communicate with each other.
- P2OS
  - Definition: P2OS, or Player/Stage Operating System, is a software framework used for controlling and simulating robots. It offers a standardized interface to control various robotic platforms and provides tools for simulation and visualization.
  - Description: P2OS simplifies the development of robot control software by offering a consistent API for different robots. It allows developers to create robot applications that are compatible with a wide range of hardware, making it a valuable choice for robot research and development.
- ArUco ID Input Publisher
  - Definition: The ArUco Id input publisher is used to get input from the user defined as an Int32 from the system.msgn.msg library. This integer is used to define the target for the pioneer robot.
  - Description: The ArUco Id input publisher is a rospy publisher node that waits for user input in the form or an integer number from the controlling system. If the user inputs any integer between negative infinity and infinity, this will be published to the 'number' topic and can be subscribed to by any node.
- ArUco Publisher Node
  - Definition: The ArUco Publisher Node is designed to utilize the camera to search for and identify ArUco markers within the frame of the camera's image output.
  - Description: The ArUco Publisher Node initiates a VideoCaptrue object from the Open-cv2 library in order to capture images from the Logitech camera attached to the RPi3. The camera is calibrated during the first steam and then continuously streams images from the camera to the node. If an ArUco marker is detected, then the marker is analyzed for pose, alignment error, distance from marker, and ArUco tag Id. These published values are then available for any node to subscribe to through their individual topics.
- Navigation Node
  - Definition: The navigation node is designed to handle all the step to escape the given enclosed environment.
  - Description: The navigation node has one main objective broken into 5 parts. It is designed to escape the environment given to and described by the Final Project Demo and Report PDF provided. It subscribes to both the ArUco ID Input Publisher and the ArUco Publisher Node. This gives it the data it needs to read the ArUco markers surrounding it to navigate and find the target marker.

III. Design

Our design uses the two sensors to accomplish this goal, the encoder and the camera. The primary event loop consists of rotating counter-clockwise until an Aruco Marker with ID of 1 is found. This marker indicates the center of the wall defined by points A and B as seen in figure 1. Our robot then centers the robot camera on the marker through use of the encoders and begins moving forward. If the marker ever becomes off

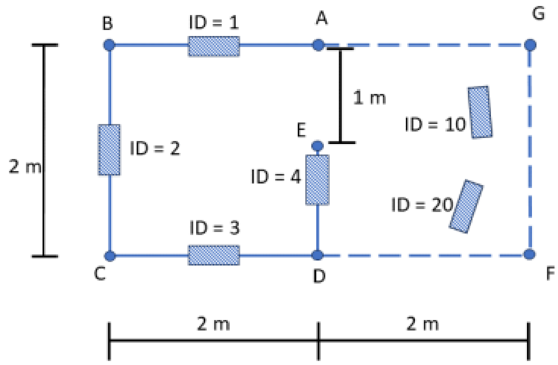

Fig. 1. Provided environment

center from the camera by more than 1 degree, the robot stops and re-centers the marker before continuing. Once the marker is within .5 meters of the camera, the robot begins rotating clockwise until either the goal marker is seen or the marker with id of 4 is seen. If the goal marker is seen the robot simply centers the marker with the camera and moves towards it until it touches, re-centering as needed. If the marker of id of 4 is found, the robot turns counter-clockwise 22.5 degrees and moves forward 1 meter with help from the encoders. Once it is at this position, the robot begins scanning back and forth until the goal marker is found and once it has been found it centers the marker with the camera and moves towards it, centering as needed.

IV. Individual Learning Outcome

CPE470 has been a captivating course that delve into robotics, discussing concepts that were entirely new to my experience. The focal point of the class was delving into the Robot Operating System (ROS), a vital component of contemporary robotic programming. Working and understanding of ROS was super interesting to learn about the way various tasks and sensors seamlessly interact in a robotic system. I thought that navigating the Pioneer Robot was a fascinating experience, especially considering my relatively limited exposure to robotics. Employing a Raspberry Pi V3 and Python within the ROS framework, I gained hands-on experience in manipulating intricate components such as wheel motors and encoders. What particularly fascinated me was the application of encoders and their pivotal role in achieving precision and control with our last project. Learning to script a publisher/subscriber to orchestrate the Pioneer Robot's movement and alignment with an Aruco marker using encoders was a standout moment. Encoders, with their ability to measure wheel rotation, provided a nuanced understanding of how precision control can be achieved in robotics. The exploration of sensors added another layer of excitement to the class. Lidar, in particular, emerged as an intriguing technology to work with. The process of converting data from polar to Cartesian coordinates and employing it for mapping our surroundings showcased the practical applications of lidar in robotics. Although there was noisy data to deal with, relying on my team to work out the sensor was a big learning curve as it showed the importance of detail-orientation within challenging projects. Beyond the technical aspects, the class offered valuable insights into collaborative endeavors. Working within a group characterized by varying levels of expertise in robotics presented its own set of challenges. Debugging, testing, and overcoming issues as a team underscored the significance of effective communication and cooperation in navigating the intricacies of robotics projects. In conclusion, this class benefitted me by broadening my theoretical understanding of robotics but also honed my practical skills in programming, problem-solving, and teamwork.

Conclusion

The successful completion of the presented navigation task within the given enclosed environment reflects the potential and adaptability of robotics and automation systems. Through the combination of sensor technologies, precise control algorithms, and strategic decision-making, our robot was able to autonomously navigate from an arbitrary starting point within room ABCD to the specified ArUco target tag, via the designated exit, accurately.

In a rapidly evolving technological landscape, the successful completion of such tasks underscores the potential for robotics to revolutionize various fields, including automation, logistics, and even search and rescue operations. The lessons learned from this endeavor will undoubtedly contribute to the continued advancement of our education in autonomous systems, making us increasingly more proficient in handling complex real-world challenges.