Mobile Robotics Challenge
Team Name: Ferrari

Priyansh Jain

June 8, 2023

1 Introduction

The world of robotics is where the cutting-edge technology meets the real world. This assignment helps us in making a mobile robot estimate and avoid an unknown object or get as close to it as possible. I used a PiCar-4wd or a RaspberryPi that controls a car. The PiCar-4wd can be seen in this picture. The

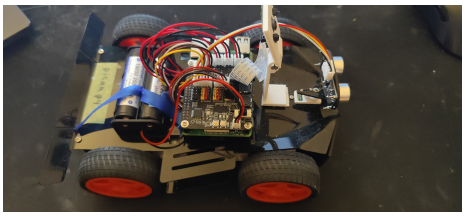

Figure 1: The PiCar-4wd

obstacles can be seen in this image:

2 Method

2.1 Challenge 1

In this challenge, we used the Picamera2 to get an image in the 1280x720 dimension as an array. Then we use the array and find the edges by grayscaling it and blurring it using the OpenCV's cv2 library. cv2 has a functions Gassianblur that smoothes images, which make it easier to find edges. Then we us Canny edge detection to find the edges which finds the edges by comparing the intensity of pixels. Now we calculate the distance to the object by using the formula $distance = (knownWidth * focalLength)/Width$ where the known width is 7 and the focal length is set to 820, the width is detected from the image. Now we move towards the object while calculating the distance to the object straight ahead. When we are in a specific range of the object then we command the robot to stop.

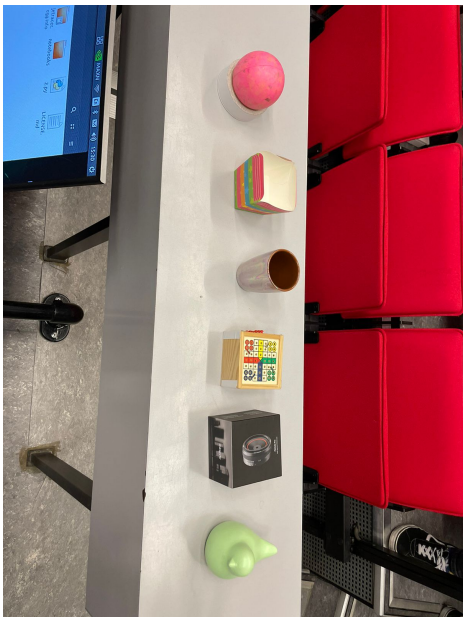

Figure 2: Caption

2.2 Challenge 2

Well, in this challenge we used the same framework as the first challenge. This time the algorithm is used on an RGB image, this is because sometimes the background may have the similar intensity as the object. The robot first starts moving forward, then when the detected object is within range, then the robot enters a loops and the moves until the obstacle is avoided. The moves are as follows: First move: Turn Right, Second Move: Move Forward, Third Move:Turn Left, Fourth Move: Move Forward, Fifth Move: Turn Left, Sixth Move: Move Forward, Seventh Move: Turn Right. The code also checks for the keyboard input to stop the process. This is how the robot avoids the object.

2.3 Challenge 3

The code imports a pre-trained model and configuration file, and a text file containing class names. The code captures an image from a Raspberry Pi camera and processes it to detect objects using the YOLOv3 model. The detected objects are then labeled with bounding boxes and class names. The code also calculates the distance between the detected object and the camera using the known width of the object and the focal length of the camera. The code then uses a Picar_4wd robot to move towards the detected object until it reaches the calculated distance. If the object is not found, the robot scans the area by turning left and right until it finds the object. The code stops when the robot

reaches the calculated distance from the object.

3 Conclusion

In this assignment, I learned how to use picar-4wd , a mobile robot to detect and avoid obstacles. The robot can detect obstacles in a straight line, find the distance to the obstacle and move right up to it. It can also detect any obstacles in a straight line and avoid them by using specific movements. Also lastly, the current version of the robot can find an obstacle that is not straight ahead of you, and move right up to it. Although the robot can only do so much now, it can be improved further using machine learning to