

# Computer Technology Project I

Simon Nyman

Dept. of Electrical and Computer Engineering (of Aff.)  
Aarhus University

Studentnumber: 202305077

Jakob Palm

Dept. of Electrical and Computer Engineering  
Aarhus University

Studentnumber: 202307244

## Abstract—

### I. INTRODUCTION

The goal for this project was to design a down-scaled version of a Search And Rescue (SAR) robot, used to rescue victims in the aftermaths of natural disasters. The robot will imitate a real world scenario by, autonomously navigating an obstacle course and distinguishing between different color markings, representing potential victims. To achieved this we use the TurtleBot3 Burger Robot equipped with different sensors. All the software will be implmented in python using the Robot Operating Software framework (ROS). For the sensors we will be using a LiDAR sensor capable of measuring distances in a 360 degree view, and an RGB-sensor which differentiates between red, green and blue colors. The performance will be assesd based on three factors: Average speed, number of collisions and "victims" found.

### II. SPECIFICATIONS

Our Search and Rescue implementation used a turtlebot3 robot, specifically the burger configuration with dimensions 138mm x 178mm x 192mm (L × W × H) [1]. On the turtlebot3, a Raspberry Pi 3 model B+ and an Arduino are connected in order to process the incoming external signals from the various sensors. Furthermore, the robot has two motors attached, one for each wheel. From the specification it is evident that the robot has a maximal linear velocity of 0.22 m/s and a maximum rotational velocity of 2.84 rad/s. In order keep the robot as agile as possible with no wires attached, we used an 800 mA Li-Po battery and established a wireless connection, which will be specified further in a following section. The specific RGB used is an ISL29125 low power, high sensitivity red, green and blue light sensor with SMBus compatibility [2]. The LiDAR used is the LDS-01 version with a range of 120 mm to 3500 mm [3]

#### A. Software Setup

1) *Network Configuration:* As mentioned, we wanted the robot to be able to move as freely as possible, meaning that connecting it to our machine via an ethernet cable was not the way to go. Therefore, we established a wireless connection to the robot by changing the network configuration on the pi. We used the following command:

2) *ROS:*

### III. METHODOLOGY AND DESIGN

To make the

#### A. Navigation

implementation

#### B. RGB

implementation

#### C. LED

implementation

### IV. EXPERIMENTATION AND TESTING

### V. CONCLUSION

### VI. DISCUSSION

#### A. Figures and Tables

a) *Positioning Figures and Tables:* Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation "Fig. 1", even at the beginning of a sentence.

TABLE I  
TABLE TYPE STYLES

Table Head	Table Column Head		
	Table column subhead	Subhead	Subhead
copy	More table copy <sup>a</sup>		

<sup>a</sup>Sample of a Table footnote.



Fig. 1. Example of a figure caption.

## REFERENCES

- [1] 'Turtlebot3 features', accessed 15 May 2024, available at: <https://emanual.robotis.com/docs/en/platform/turtlebot3/features/>
- [2] 'Renesas RGB-sensor ISL29125 datasheet', accessed 15 May 2024, available at: <https://www.alldatasheet.com/datasheet-pdf/pdf/1045936/RENESAS/ISL29125.html>
- [3] 'LDS-01 overview', accessed 15 May 2024, available at: <https://www.robot-advance.com/EN/art-360-laser-distance-sensor-lds-01-2352.htm>