Simon Fraser University
School of Mechatronic Systems Engineering
MSE110 Mechatronic Design I

PROJECT 1 – Line Tracking

Demonstration Date: Tuesday, January 29th, 2019

Overview

Line following robots are widely implemented in industrial robotics because they allow the robot to move from one point to another and perform various tasks. A line tracker employs light sensor(s) to follow a line and depending on the color of the path, it is expected to do different reactions upon facing an obstacle in their path. These sensors can be put in various layouts to improve the performance of the robot (Figure1).

In this project, the goal is to meet the tasks **as fast as possible, and without losing accuracy.**

Light Sensor Calibration

The first step is to define an intensity threshold to distinct what it is considered to be a line from the background. Therefore, the light sensor should be calibrated for this purpose. We need to measure reflected light when we use the sensor in Reflected Mode. The reflected light value is interpreted as a number between 0 and 100 according to the amount of light the sensor is receiving back. You can hold the sensor once on top of the white surface, and once on top of the line (blue or green) surface and read the sensor output values for each of the surfaces and use these numbers to define thresholds for programming.

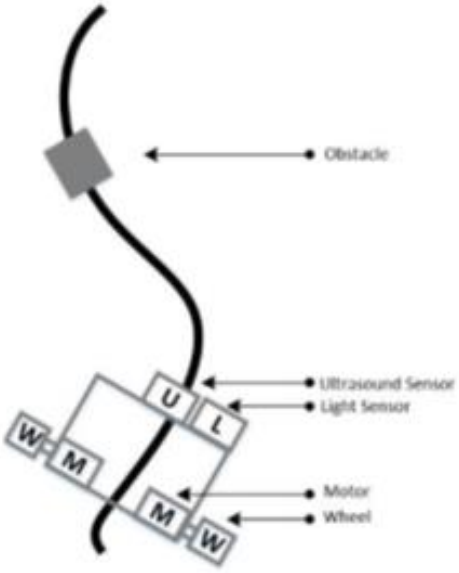

Figure 1- Schematic of the line follower robot

Line Following with On/Off Controller

We place the robot on one side (left/right you may decide) of the line and let it read the light sensor. Assuming that you know if the line is in the right or left-hand side of the robot, you may design a proper strategy to follow the path accurately. If the light intensity reflected from the surface was below a defined threshold, the robot should approach the line, and if it is above the threshold that means the robot touch the line and it should move to the opposite direction so that it does lose the path.

To make a turn while moving forward, one wheel should rotate faster than the other. For example, if the robot is programmed to turn left, the right wheel should rotate faster than the left one.

Task Description

Design a Line-Tracker robot to perform the following:

a) Track the simple line (given an initial condition). You can decide if the robot is initially in the left-hand side or right-hand side of the line (Figure 1).

b) Once an obstacle is seen in 10 cm distance, the robot should stop and make a beep for two seconds.

c) According to the color of the line, the robot should do one of the following tasks.

If the line is blue:

1. the robot continues moving toward the obstacle and pushes it to the left or right, at least 10 cm (roughly), away from its original position and in the direction where the obstacle will be away from the line.
2. return to the tracking line and continue moving forward.

If the line is green:

1. it should turn 180 degrees and move backward to the end of the path.

d) Your robot must be able to follow the line when it reaches a path with maximum 90 degrees sharp turns.

<u>Bonus points</u>:

You will be able to get extra marks for adding each of the following abilities to your robot:

a) Design a PID controller for Line Tracker to follow the line faster and in a smoother way.

b) The robots employed in industries may face some (unexpected) complexities while following their line. These changes include the shapes represented in Figure 2. To get the bonus points, the robot may be able to follow the line with sharp edges with less than 90 degrees (but not less than 30 degrees). You do not need to track any other changes presented in Figure 2.

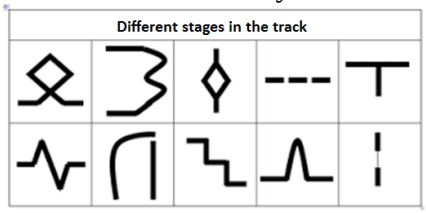

Figure 2- Examples of complexities that may occur in the tracking line in real-life situations

Marking criteria

As mentioned in section c of the task description, your robot should be able to do both tasks described in c.1 and c.2. We will test the robot in both scenarios. More specific details for marking are given below.

Criteria	Mark out of 100	Description
Mechanical Design	10	Weight distribution, robustness, proper position of the sensors.
Smoothness of the movement	20	Smooth continuous movement without skidding.
Duration	15	Marks for this criteria will be normalized based on the best and worst performance among all groups.
Stop 10cm before the block	5	Robot should completely stop when it sees the block.
Two seconds beep 10cm before the obstacle	5	Make a beep for two seconds.
Correctness of the color sensing	10	will be marked based on the consequent action of the robot after recognizing the block.
Removing mechanism of the block	15	Mechanical design and algorithm used to remove the block.
Being able to follow 90 degree turns	20	-
Using PID in the algorithm	Bonus (+%5)	-
Being able to follow 30 degree turns	Bonus (+%8)	-