JACOBS
UNIVERSITY

Autonomous Duckie(Jacobs_Duckie) Line Detection and Following

by

Santosh Iuitel, Deepak Budha, Uriel Pauline Jimenez Palmos

SPRING 2022
CO-548-A RIS Project

Submission: August 7, 2022 Supervisor: Prof. Dr.Amr Alanwar Abdelhafez

Jacobs University Bremen | Department of Computer Science and Electrical Engineering

Abstract

One of the most crucial features of robotics is line following. A Line Following Robot is an autonomous robot that can follow either a black line placed on a contrasting colored surface or a white line. It is programmed to move automatically and in accordance with the line. This project intends to apply the algorithm and control the robot's movement through proper tuning of the control parameters, resulting in improved performance. It can be utilized for industrial automated equipment transporters, small residential applications, museum tour guides, and other similar applications, among others.

Keywords: Line detection

Contents

1 Introduction

A line follower is a machine that can follow a path. The path can be visible like a black line on a white surface. Sensing a line and maneuvering the robot to stay on course, while constantly correcting wrong moves using feedback from the sensor forms a simple yet effective system. It can be used in automobiles, industrial automation, guidance, etc

1.1 Background

As technology becomes increasingly important in today's world, it is invaluable to not only learn how to use technology but also to understand how to create it. Since being an engineer one should have sound knowledge of the other discipline. Most of the projects have limited scope to only specific disciplines. This would limit one's innovation and creativity. This project inspires me to make connections across several disciplines rather than learning topics in isolation as it combines mechanical, electronic, electrical, and programming skills. The Duckie robot is designed in a such way that it tracks the path and follows it.

1.2 Motivation

How ants always travel in a line, following an invisible route in search of food, or back home. How on roads the lanes is followed to avoid accidents and traffic jams. Ever thought about a robot that follows a line? A perfect or near-perfect mimic of nature? After all, the purpose of robotics is to recreate in terms of machines what one

sees around to solve a problem or fulfill a requirement. The area will be benefited from the project

- Industrial automated equipment carriers.
- Entertainment and small household applications.
- Tour guides in museums and other similar applications.

1.3 Problem Statement

In the industry, carriers are required to carry products from one manufacturing plant to another which are usually in different buildings or separate blocks. Conventionally, carts or trucks were used with human drivers. Unreliability and inefficiency in this part of the assembly line formed the weakest link. The project is to automate this sector, using carts to follow a line instead of laying railway tracks which are both costly and an inconvenience.

1.4 Objectives

The objectives of the project are:

- The robot must be capable of detecting a line
- The robot must be able to navigate while following a straight line
- Stop signal when the straight line is out of sight

2 Implementation

2.1 Solution Approach

We are using the Hough Transform algorithm to detect a straight line. We are using the implementation from CV2 to detect the straight line in the camera setting. Considering the object detection and vision part of the Duckiebot, we are taking a black line placed on a contrasting colored surface or a white line that can be detected and making the Duckiebot follow the line as long as it can see it.

2.2 Algorithm Used: Hough Transform

Hough transform is one of the feature attraction methods popularly used for line detection within a certain class of shapes with a voting method. It is important to reduce the size of data while preserving its important aspects of it. This can be done by edge detection. Initially, Hough transform was only used to detect lines but now it is popularly used to detect arbitrary shapes like circles. The way Hough transform work is by changing the parameter space from point to its slope and y-intercept. It changes the xy space to mc space where m is the slope and c is the y-intercept. The simplest case of the Hough transform is the linear transform for detecting straight lines. In the image space, the straight line can be described as y=mx+c and can be graphically plotted for each pair of image points (x, y). In the Hough transform, a main idea is to consider the characteristics of the straight line not as image points x or y, but in terms of its parameters,

here the slope parameter m and the intercept parameter c.

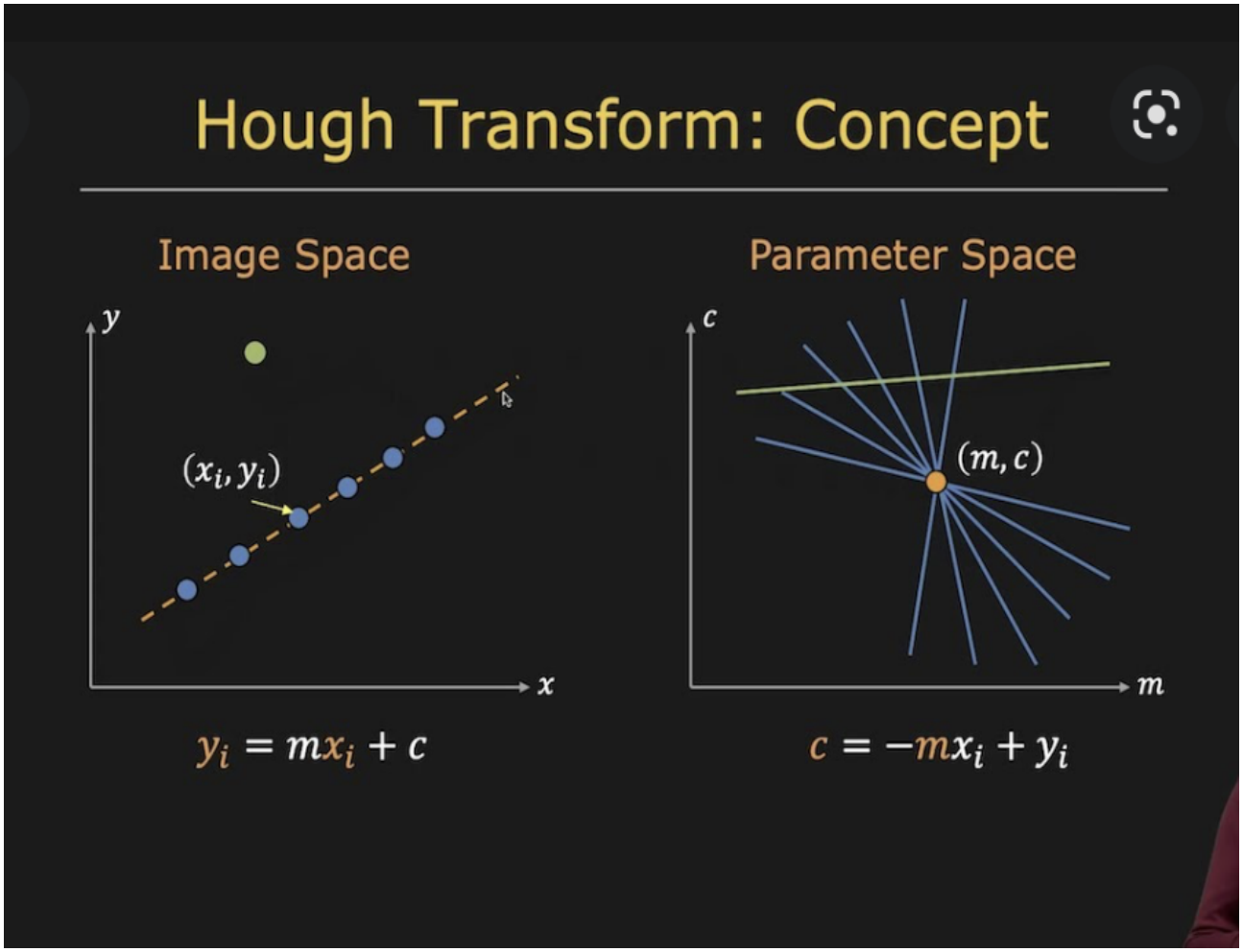

Figure 1: Hough Transform : concept

2.3 Implementation of the solution(Code)

Our approach in implementing the idea of making the Duckiebot follow the line is basically divided into two parts

2.3.1 Line Detection

Utils.py script contains the code for the algorithm used in this project. We used python in the implementation of the Hough Transformation and the execution and testing as well, attached are the images of Hough's concept and

the snippets of the implemented code.

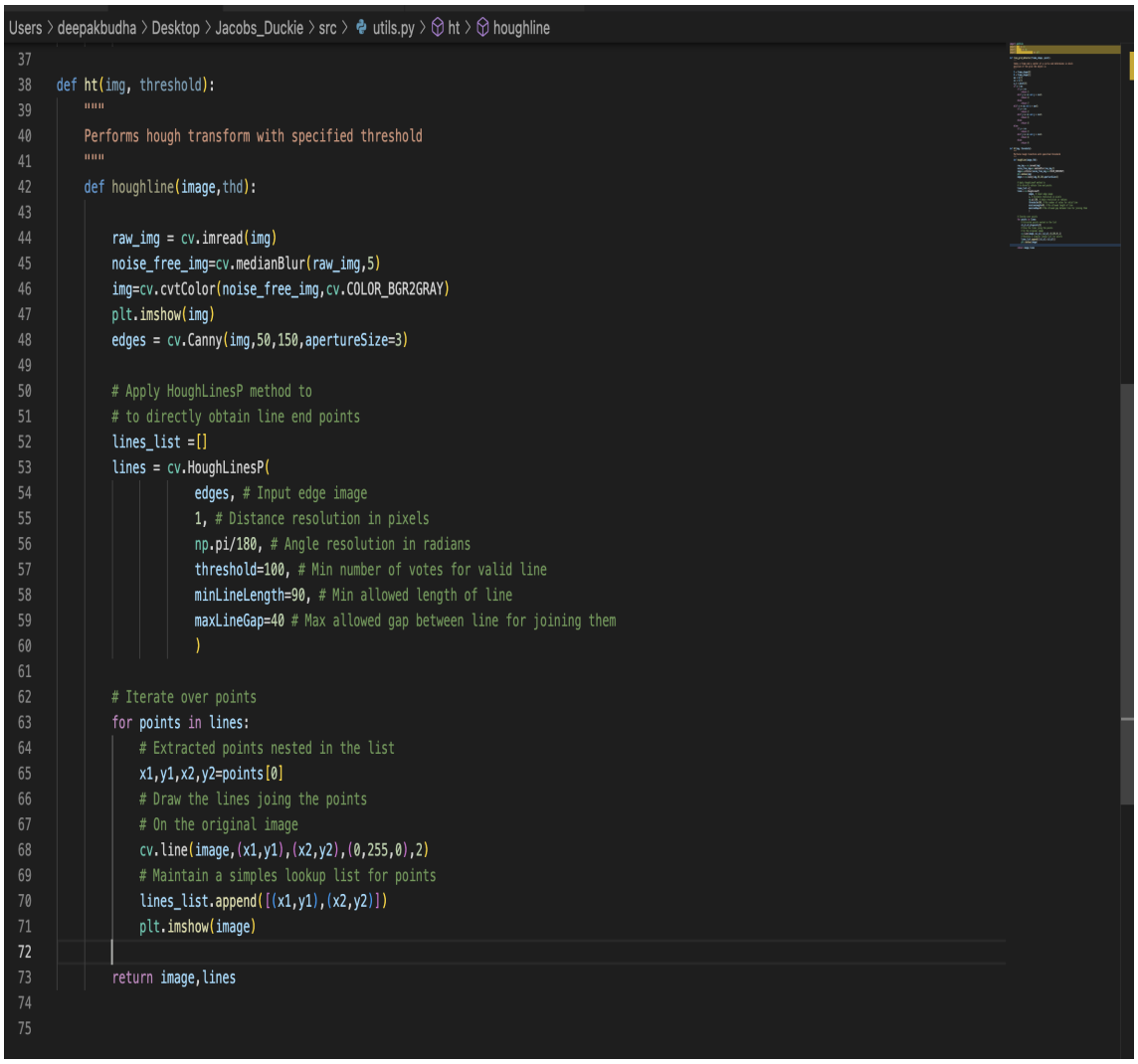

Figure 2: snippet of util.py

With the Arguments: Img: Applying median blur in order to reduce noise converting it to grayscale edges: detects edge to reduce to the size of the image to make it easy for computation lines: gives us the coordinates of the line threshold: min number of votes for the valid line minLineLength: min allowed length of line maxLineGap: max allowed the gap between lines for joining them

Below can be seen an images where our algorithm has detected the straight line.

Figure 3: detected line

2.3.2 main.py

main.py script contains the code for the publisher and subscriber class implemented that enables access of the wheel to follow the object (publish) as soon as it subscribes to the image view. We have the subscriber class that gets the input parameter of Compressed Image followed by a callback function for determining the grid of the object with the help of function determining the grid of the object with the help of function we take from utils.py script. After it gets the position as a grid, it moves and

stops as a certain condition is provided.

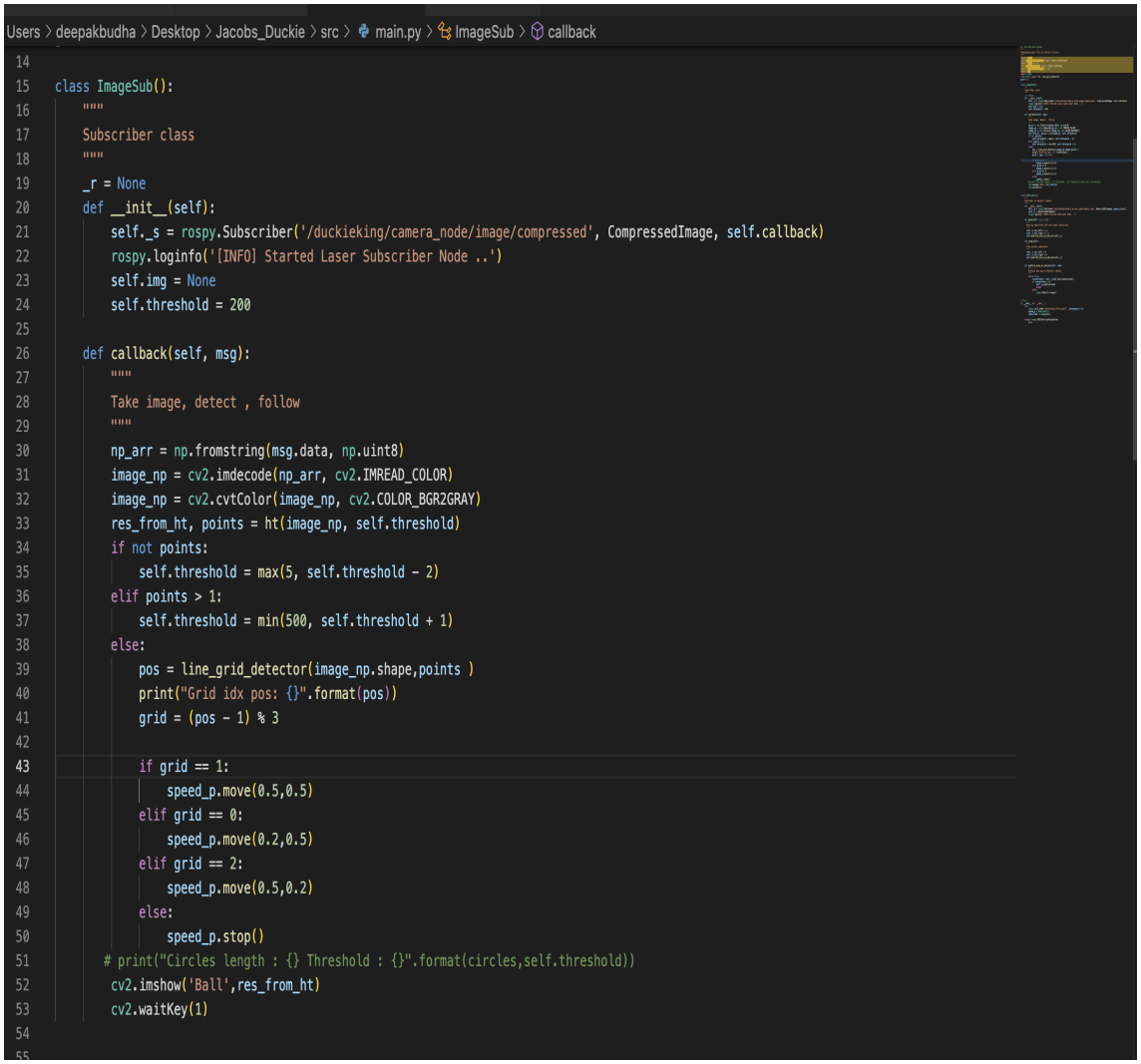

Figure 4: snippet of main.py

Finally, we have integration or the data transfer to the wheel in order for it to react with the publisher to duckie's wheel. This basically is the node for the Duckiebot to integrate the input to the backend to have the wheels command running as an output. We have functions for the specific speed of the left and right wheels. So, given the velocity of the movement = 0, the wheel stops so does the Duckiebot.

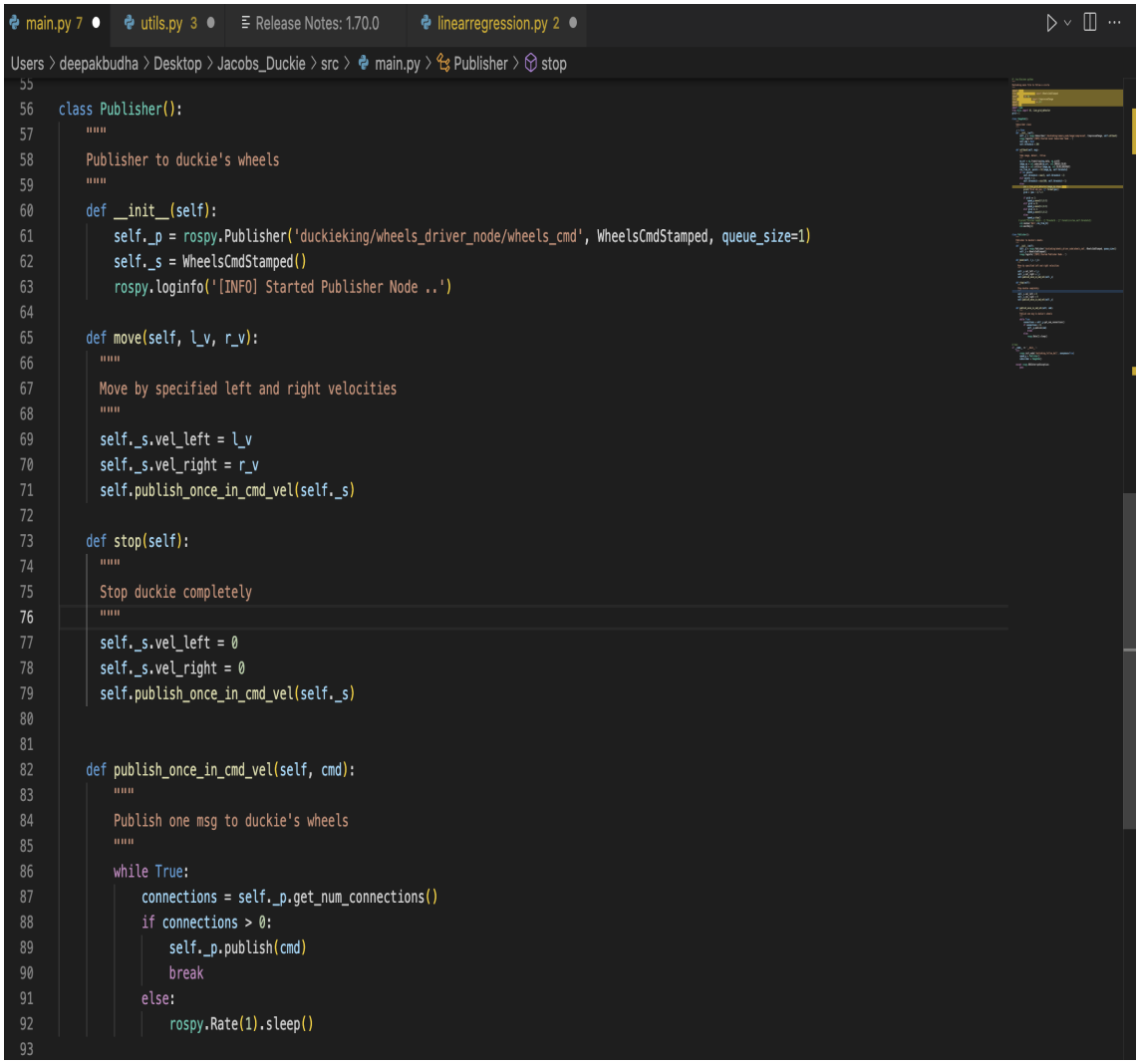

Figure 5: code snippet

We have the 'duckietown_mgsg.msg' in the folder with main.py from where ROS2 message and service definitions are used in Duckietown.
REF: https://github.com/nlimpert/duckietown_msgs

3 Future Application

The major application of this would be the self-driving car, which is still in upgrading progress so, so this concept can be certainly used to improve self-driving cars. This kind of feature can be used for industrial robots to check the warehouse or store for storing twin databases of the store and manage the logistics more efficiently. Several home base appliances like cleaning robots and other commercial applications could be imagined. It can also be used in museums as well as in several commercial applications where it can be used/Another major application would be in the agricultural sector.

4 Conclusion

The line hough method implementation may therefore be used to identify the line in noisy or poor photos as a solution to numerous difficulties and has potential growth in various sectors. The solution, as simple as it appears, has more complex ramifications in practice and would take more than three months to create and implement in a stable Python environment with the highest efficiency. This model, however, may be improved by putting greater focus on visibility and remote image capture. It is simple to run by following the instructions in the 'readme'.

5 Future Work

In the process of development of the line follower, most of the useful features is identified and many of them were implemented. But due to time limitations and other factors, some of these cannot be added. So the development features in brief:

- Use of color sensor.
- Use of better CCD camera for better recognition and precise tracking of the path.