Project Report: Collision Avoidance using Lane Detection and Object Detection
Authors: Emmanuel Mutuku & Irfan Karmali

Introduction

This project aims to implement collision avoidance using a combination of lane detection and object detection. The project utilizes a robot called robotLA. The collision avoidance system is designed to detect lanes and objects (yellow ducks) in the environment and make decisions based on this information to avoid collisions and accidents.

Code Overview

The code provided is written in Python and consists of several modules and libraries. Here's a step-by-step explanation of the code and its functionality:

1. Importing Required Libraries:

We import necessary libraries such as `os`, `cv2`, `numpy`, and `rospy`. These libraries provide various functionalities required for image processing, object detection, and interaction with the robotLA.

2. Object Detection Section:

In this section. we import the necessary modules and functions for object detection using a pre-trained model thanks to roboflow. The object detection section uses the YOLOv5 model for detecting objects in images. It also performs non-maximum suppression and applies a classifier to the detected objects. The final image with object detection results is obtained using the `detect` function.

3. Lane Detection Section:

In this section, we import modules and functions related to lane detection, including `canny_func` and other utility functions. The lane detection process involves converting the image to grayscale, applying Gaussian blur, and detecting edges using the Canny edge detection algorithm. We then identify the region of interest and detect lines using the Hough transform. The detected lines are averaged and displayed on the image via green lines as observed in the video.

4. Detector Class:

The code defines a class called `Detector`, which serves as the main component for collision avoidance. It initialises with the provided options and sets up a publisher to publish the processed image. The class includes a callback function that receives raw images from the robot's camera. Inside the callback function, the line detection function is called to process the received image and obtain an image with detected lanes. Finally, the processed image is published using the ROS (Robot Operating System) framework.

5. Main Function:
This section initializes the ROS node and sets up the argument parser for configuring the object detection parameters. It checks the requirements and creates an instance of the `Detector` class. The subscriber function is called to receive raw images from robotLA's camera. The program continues to run until an interrupt occurs. A ROS Interrupt specifically.

Conclusion:
In conclusion, this project report provides an overview of the code implementation for collision avoidance using lane detection and object detection on the RobotLA (Duckiebot) platform. The code combines the functionalities of object detection with YOLOv5 and lane detection using the Canny edge detection algorithm. By integrating these techniques, the system can detect lanes and objects in real-time, enabling the robot to make informed decisions to avoid collisions and accidents in real-world scenarios.