Autonomous Vehicles Semester Project Report

Authors:

Garrett Barton, Michael Belanger, Samuel Maloney, and Vishaldeep Singh

Professor: Dr. Supratik Mukhopadhyay

Course: ENGR 4200 Autonomous Vehicles

Submitted on: 201800427 ( pages)

**Abstract:** The team was tasked with designing and creating a system to be able to attach to a small, gas powered golf cart, and allow said cart to be able to drive autonomously around the university campus. The system was to be able to detect pedestrians, stop safely, and navigate autonomously end-end. The team was able to achieve the safety, detection of pedestrians, and the navigation goals tasked with.

**Objective**

The objective of this project was to design and create a system to allow a golf cart autonomously drive around the university campus. The main focus of this project was the software and electronics surrounding autonomous driving. The stages of the project will be detailed below.

There were several specific requirements that students needed to meet in order to have a successful project, in terms of the scope of this project. Those included being able to: capture real-time location information, detect pedestrians and stop for them, and manipulate the drive systems of the cart to be able to control the cart while driving. The stressed objective by the professor was that students gain an in-depth exposure to autonomous robotic systems. The evaluation of the project was to be weighted towards learning not milestones. The team was given a semester to independently work towards this goal, totaling to around 40-45 hours of in class time.

**Summary of Procedure**

The team began the process by first attempting to capture real-time location information. This was attempted using an Andriod phone, which would have utilized Google Maps services. However, the phone could not grab accurate information reliably, and so could not be used in the project. The team then decided to use all Raspberry Pi devices, as there were already specific equipment for such tasks. The team used a GPS module and an 8 megapixel camera compatible with the Raspberry Pi. As for path planning, the team decided to utilize a simpler, self-made algorithm. This algorithm’s basic idea is to…

After obtaining current location info, the goal shifted to being able to detect pedestrians. This was researched and several informative sources were utilized. The two most helpful, other than tutorials for downloading the software onto the Raspberry Pi, were: a blog, *PYImageSearch*, and a GitHub account, *galenballew*. Both were run by experienced coders that specialized in computer vision, and provided detailed discussions of the implementation and application of a pedestrian detection system. Their work was instrumental and instructional in completing this project.

At this stage testing needed to be done to ensure that the system could safely work if implemented on a vehicle. The team had to show that the system would be able to: detect any pedestrians, to send a signal to stop the vehicle, and navigate through road ways. Once the team could demonstrate these abilities, the team was allowed to implement this system onto the golf cart to be used. The electronics were mounted appropriately and a system to manipulate the steering wheel and peddles was created.

**Subsystem Details**

**Current Location**

The Adafruit Ultimate GPS Breakout Version 3, with 66 channels and 10 Hz updates was chosen due to its compatibility with the Raspberry Pi. The device came factory ready to be able to pull current coordinates via any GPS available. The code that the factory instructed to install, pulled data and delineated the data into easily usable forms.

**Navigation Algorithm**

fff

**Pedestrian Detection**

The pedestrian detection was implemented utilizing the 8 megapixel, Raspberry Pi camera board v2 to provide the video feed needed to run the pedestrian detection algorithms. As with the GPS module, the camera was designed to work with the Raspberry Pi and so implementation was easy. The camera is itself is a plug-and-run system, and the image capturing was very simple. The difficulty came when images needed to be sampled from the video to be processed to detect pedestrians. There were several complications with this subsystem. The first complication that the team ran into was getting a single image from the video feed to sample. This was eventually solved via… The second complication that the team found was the time requirements of taking a sample image, processing it, and then returning the necessary commands to control the actuators. To optimize this process the team tested the run time by adjusting the resolution and the sample rate. After testing the team found that the best ration between the resolution and the sample rate was…

**Issues Dealt with**

The Android phone using Google Map services

Computer vision optimization of characteristics

**Conclusion**

sfhsdh

**Resources**

Rosebrock, Adrian. “Pedestrian Detection OpenCV.” PyImageSearch, 7 Sept. 2016, [www.pyimagesearch.com/2015/11/09/pedestrian-detection-opencv/](http://www.pyimagesearch.com/2015/11/09/pedestrian-detection-opencv/).

Ballew, Galen. “SDC-Lane-and-Vehicle-Detection-Tracking: OpenCV in Python for lane line and vehicle detection/tracking in autonomous cars.” galenballew, 24 April 2017, <https://github.com/galenballew/SDC-Lane-and-Vehicle-Detection-Tracking>.

**Materials**

* Raspberry Pi 3
* Adafruit Ultimate GPS Breakout - 66 channel w/10 Hz updates - Version 3 (Product ID: 746)
* Raspberry Pi Camera Board v2 - 8 Megapixels (Adafruit Product ID: 3099)
* Flex Cable for Raspberry Pi Camera or Display - 2 meters (Adafruit Product ID: 2144)