

# Final Capstone Problem Statement

GFR Group

CS461 Senior Software Engineering Project

Fall 2018

December 4, 2018



Autonomous vehicles require robust representations of the world in order to make correct driving decisions. The driverless vehicle must utilize sensor input from lidar, camera, accelerometer and GPS in order to construct a virtual representation of the world around it as well as localize itself within that world. Due to the inherent error and low sampling rates of the sensors, high speed motion can cause data about the environment to be sparse and noisy. Robust statistical analysis is needed to evaluate the confidence of the vehicle's internal representation of the world around it. Our team is tasked with constructing the vision, mapping, and localization systems for a driverless formula style racecar for the Global Formula Racing (GFR) team. The driverless vehicle is expected to compete in the Formula SAE (FSAE) competition in August 2019.

## CONTENTS

<b>1</b>	<b>The Problem</b>	<b>3</b>
<b>2</b>	<b>The Solution</b>	<b>3</b>
2.1	General Description of Approach . . . . .	3
2.2	Computer Vision . . . . .	3
2.2.1	Visual Vision: Cone Detection with Convolutional Neural Network . . . . .	3
2.2.2	Laser Vision: Environment Segmentation and Cone Detection with Lidar . . . . .	3
2.3	Simultaneous Localization and Mapping (SLAM) . . . . .	4
2.4	Sensor Statistical Analysis . . . . .	4
<b>3</b>	<b>Performance Metrics</b>	<b>5</b>
3.1	Computer Vision . . . . .	5
3.1.1	Evaluating Cone Detection Performance . . . . .	5
3.1.2	Cone Detection Using Camera . . . . .	5
3.1.3	Cone Detection Using Lidar . . . . .	5
3.2	Simultaneous Localization and Mapping (SLAM) . . . . .	5
3.3	Sensor Statistical Analysis . . . . .	6

## 1 THE PROBLEM

The driverless vehicle will compete in 4 types of competitions: skidpad, straight-drive, autocross, and trackdrive. In skidpad, the vehicle must complete several laps around a figure-eight style track. In straight-drive, the car must accelerate in a straight line as fast as possible. In autocross, the vehicle must complete one lap around a custom track as quickly as possible. And in trackdrive, the vehicle must complete 10 laps around a custom track as fast as possible.

The driverless vehicle must take advantage of sensor readings from the cameras, lidar, accelerometer, GPS and other sensors and turn them into a virtual representation of the world. Blue and yellow traffic cones make up the left and right boundaries of the track and are set up differently depending on the type of competition. In trackdrive and autocross, the vehicle must map its environment, the track. For the other two events, acceleration and skidpad, the track is known and so does not have to be dynamically built through sensors. The vehicle must also be able to identify its state within the environment, which will include its position, ground speed, heading, angular velocity, and more.

## 2 THE SOLUTION

### 2.1 General Description of Approach

Our approach to object detection is to use 2 types of sensors: lidar (laser scanner) and camera. Both of these sensors will independently and simultaneously collect data about the surrounding environment. From this data, objects will be detected, distances measured, and finally used in the sensor fusion component where data from lidar and camera are combined, calibrated, and made into one result with a degree of certainty. Finally, this result will be sent to our Simultaneous Localization and Mapping (SLAM) component. Based on the data received from the sensor fusion component, this component will simultaneously, while the car is moving very fast, estimate where the car is with respect to the objects around it and to the whole racing course.

### 2.2 Computer Vision

#### 2.2.1 Visual Vision: Cone Detection with Convolutional Neural Network

Our vision system must utilize some form of the Simultaneous Localization and Mapping (SLAM) algorithm in order to keep track of how the world changes around the vehicle as it moves. To do this, information from lidar, RGB camera, accelerometer, GPS, and other sensors must be used to construct a robust representation of the world and establish where the vehicle is in that world. Single Shot Multibox Detector, Faster-RCNN (Region Convolutional Neural Network) and YOLOv3 (You Only Look Once version 3) are powerful neural networks for detecting objects. One of these networks can be used to find and classify the blue and yellow cones.

#### 2.2.2 Laser Vision: Environment Segmentation and Cone Detection with Lidar

The lidar is a laser scanner that works fundamentally like a radar but uses light from a laser instead of pulses of high-frequency electromagnetic waves. The lidar sends out pulses of light that hit the objects around and then reflects off these objects back to the lidar.

From the reflections of laser light, we get a set of 3D data points, so called 3D Point Cloud from now on, that figuratively represent the physical environment surrounding us at the moment. With this 3D Point Cloud, we filter out the data points of objects that are outside of the racing track such as walls, buildings, human, and lastly the ground because we are not concerned about the ground of the track. The result of this filtering are the light intensity of the cones - the object that we care about - and most importantly, their positions with respect to the car.

### **2.3 Simultaneous Localization and Mapping (SLAM)**

A large problem that autonomous cars have is understanding a new environment and knowing where it is located within that environment. This is the problem SLAM solves. SLAM takes information from the sensors on the car and recursively calculates the most probable location of track features and car location. There are a few common ways to accomplish this, one of which is the Kalman Filter.

Kalman filters are common tools for filtering uncertain sensor data and estimating the real value of a measured quantity. Simple Kalman filters are designed for linear data, while extended Kalman filters will be necessary for nonlinear measurements, though both have the same interface once parameterized. Kalman filters are commonly used in robotics and have many open source implementations in ROS, therefore most new development for this project will be in the domain of sensor fusion.

By using the Kalman Filter we are able to algorithmically figure out the tracks path. The Kalman filter takes in several data points given to it by the computer vision software and dynamically estimates future points. Kalman filters have been proven to be extremely effective with other robotics in areas of planning, motion, and control. Furthermore, it is useful for real time data analysis.

In SLAM, recognizing a place that has already been visited is referred to as loop-closure detection. Such a detection makes it possible to increase the precision of the pose estimate and global localization results by correcting the problem of error accumulation. A very important challenge in loop closure detection is that the method should not give us false positive results, simply because too false positive results can cause a total failure of our mapping procedure. Another method we consider is FastSLAM, a particle filter based method. To detect the closing of the race track, a simple finite state machine method is used. That is, wherever the vehicle goes, it maintains the state of the world as a particle. Every particle has 3 states, Initialized, TravelledAway and ReturnHome. Particles start at an Initialized state and after certain amount of travel, they are transit to TravelledAway state, and the ReturnHome state would be triggered when it travels back within a certain amount of the starting point, with a heading not deviating more than some threshold angle from the starting heading. When all particles have reached the ReturnedHome state, the map is complete and the vehicle can begin devoting more resources to localization and less to mapping.

### **2.4 Sensor Statistical Analysis**

To ensure sensor accuracy and detect faults or disturbances, robust statistical analysis and fusion of sensor data will be used alongside a mathematical model of the vehicle to refine hardware measurements with Kalman filter. Estimated values and known uncertainty will also be necessary to compensate for the asynchronous

nature of the vehicle's sensors. The development in sensor fusion will require using the mathematical model to calculate uncertainties and determine the full estimated state of the vehicle.

Because the vehicle's sensors operate asynchronously, sensor data is gathered at different rates and sensor readings are often offset from one another. To properly determine the vehicle state at any time step, a feature necessary for localization, sensor data will be interpolated through time using control inputs and the previous vehicle state. This is a nontrivial problem, as some constraints of the vehicle are nonlinear and so would violate expected constraints if naively projected into the future. The vehicle's mathematical model and the estimated state as provided by the sensor filtering will be used to project the full state through time.

### **3 PERFORMANCE METRICS**

#### **3.1 Computer Vision**

##### *3.1.1 Evaluating Cone Detection Performance*

We obtain ground truth to evaluate our performance of our localization and mapping by using a simulated environment (GazeBo is one potential candidate). The ground truth is that we know all the information about the true locations of the cones. We compare this true location with the detected location produced by our algorithms for lidar and camera.

##### *3.1.2 Cone Detection Using Camera*

In order to validate the vision system's accuracy, simulations of our vision system will be run and the location and count of traffic cones in our virtual reconstruction will be compared to the count and location of the traffic cones in the simulation. Detection speed is also an incredibly important factor. The vehicle will be moving as fast as 60 miles per hour or 88 feet per second. Our vision system should ideally run at a minimum of 88 frames per second, or 1 frame per foot traveled in order to have enough time to correct for errors in the trajectory. Additionally, the false positive and false negative cone detection detection rates should be no greater than 5% of all cones detected during the operation of the vehicle.

##### *3.1.3 Cone Detection Using Lidar*

The performance evaluation metrics of cone detection with lidar are very similar to those of camera: the error rate of the detected position compared to the actual ground-truth cone positions we put in our simulation. In addition, we are also concerned with how many cone we missed and whether we included other false objects in our detection. So, Hit Rate (%) and False Position (%) are two additional metrics.

#### **3.2 Simultaneous Localization and Mapping (SLAM)**

Estimating SLAM's performance metrics is challenging since it relies heavily on the accuracy of the data we receive from lidar and camera processing. Measuring how closely our mapping resembles the actual track demonstrates the accuracy of our predictions and thus the accuracy of SLAM. Additionally, our system should be able to function properly even if a sensor is not functioning or is producing bad data.

### **3.3 Sensor Statistical Analysis**

The most basic requirement of proper sensor measurement is convergence and stability. The estimated vehicle state should approach the true vehicle state over time and should not reach instability and allow any variable to grow towards positive and negative infinity. The estimated state should be accurate within a 95% margin after full convergence.

Because there are redundant sensors on the vehicle, it is important to design the measurement of the vehicle state to be resilient to individual sensor failures or disturbances. Failure of a single sensor should not have a significant impact on the overall performance of the vehicle. If multiple independent sensors fail, the vehicle should still be able to complete the track at a reduced speed.