Design, Control, and Optimization of Autonomous Racecar

LITESPEED
Smart, Swift, Simple

David Liu | Ee-Ning Ooi | Kim Dang | Max Ding | Sean Burke

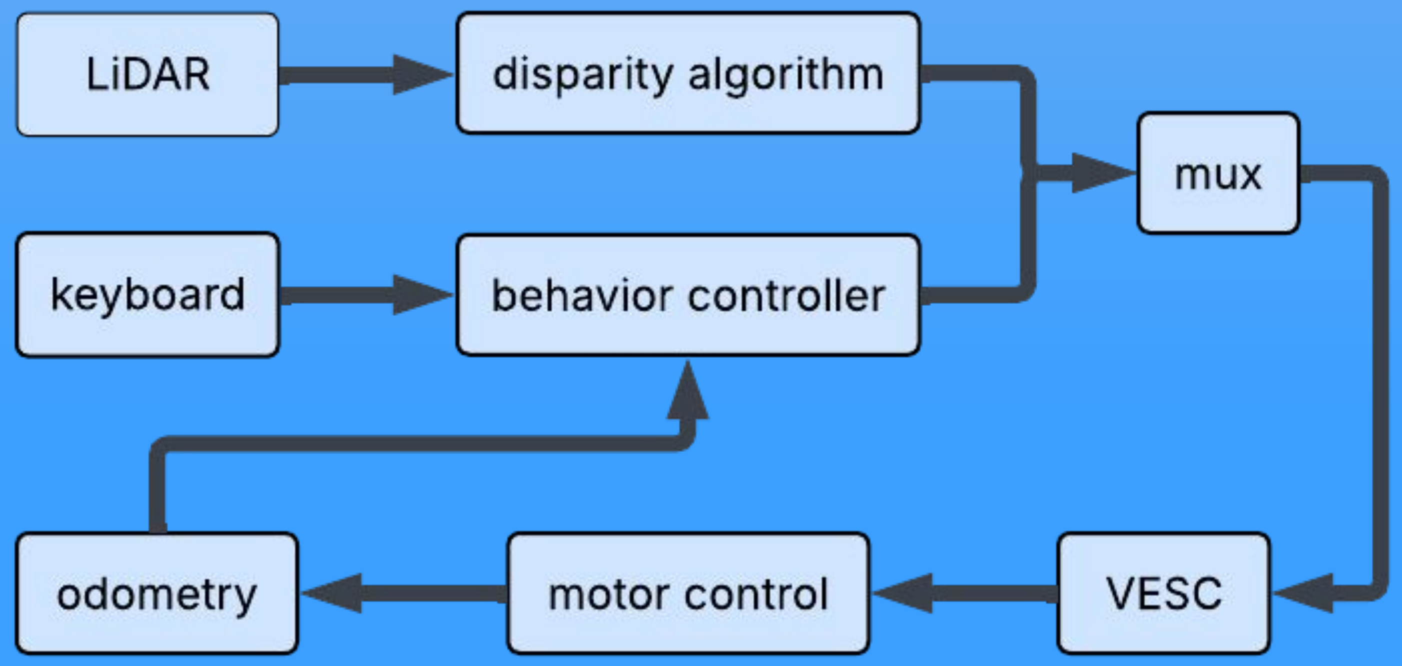
Background

The emerging technology of driverless vehicles promises to revolutionize transportation and elevate safety, efficiency, and convenience. Our project builds upon a foundational platform for exploring hardware and software systems integral to self-driving vehicles. The goal for our project was to develop an autonomous 1/10 scale car to drive around any racetrack as quickly as possible without crashing.

Our Solution

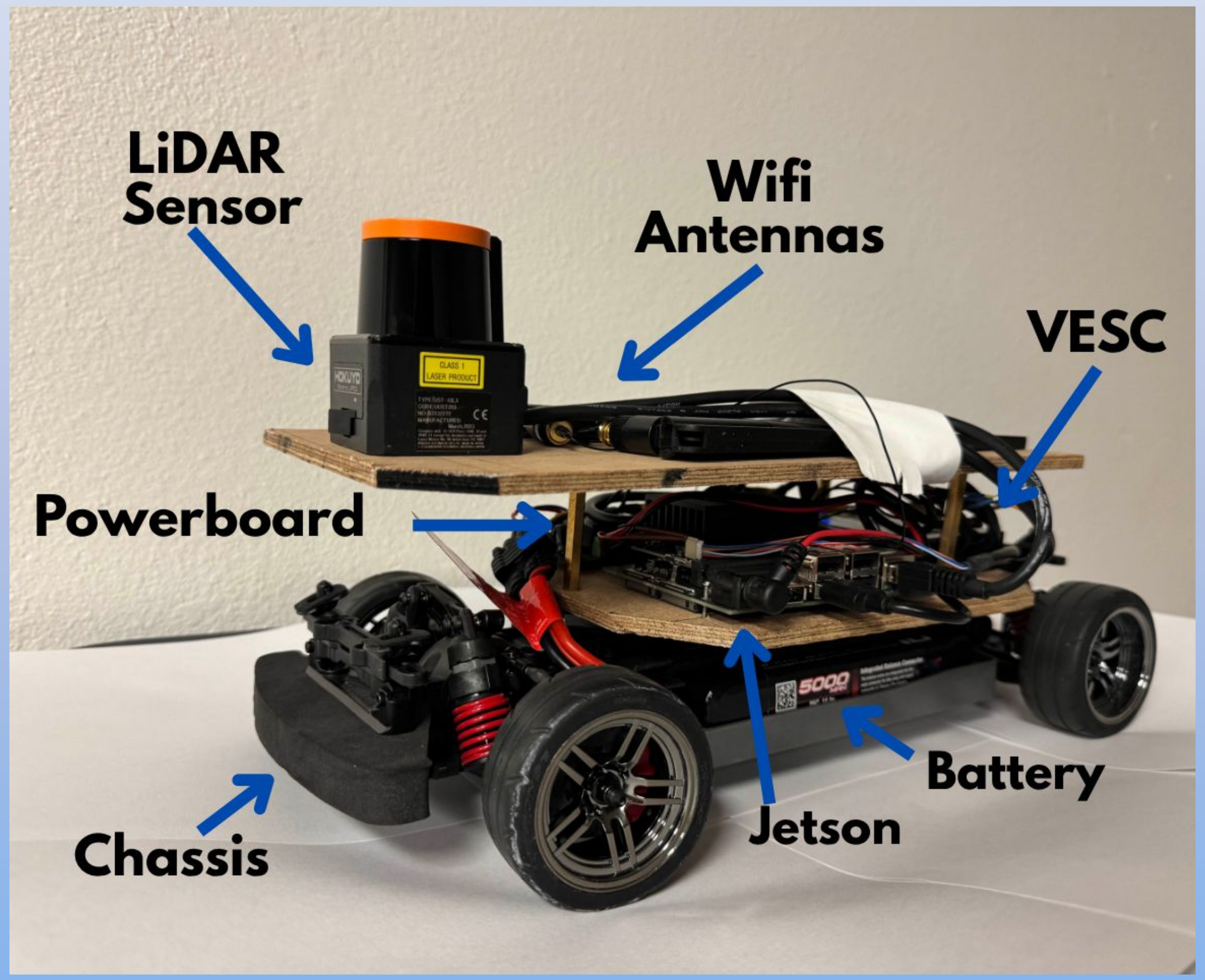
Our car navigates using an implementation of the Disparity Extender algorithm, accelerating to high speed when the path ahead is clear and slowing down as a corner or obstacle approaches. A LiDAR sensor running at 40 Hz receives measurements of distances from obstacles. WIth each sample of data, the algorithm finds the farthest direction it can safely travel while accounting for car width and speed. The car steers itself towards that direction until a new LiDAR sample arrives and the process repeats. The car's speed changes depending on distance, linearly scaling with the length of the clear path ahead. This algorithm was implemented in C++ and integrated with ROS.

Robot Operating System (ROS)



The ROS framework uses a system of **nodes** that either **publish** or **subscribe** to information. Pictured above is a greatly simplified map of our how our code connects together in ROS.

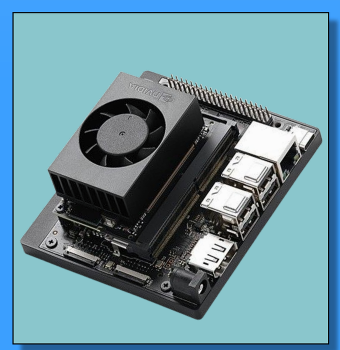
Final Car



Key Components

Traxxas 4-Tec 2.0 VXL
Chassis with a brushless motor, 9 mm ground clearance, and a top speed of 70 mph

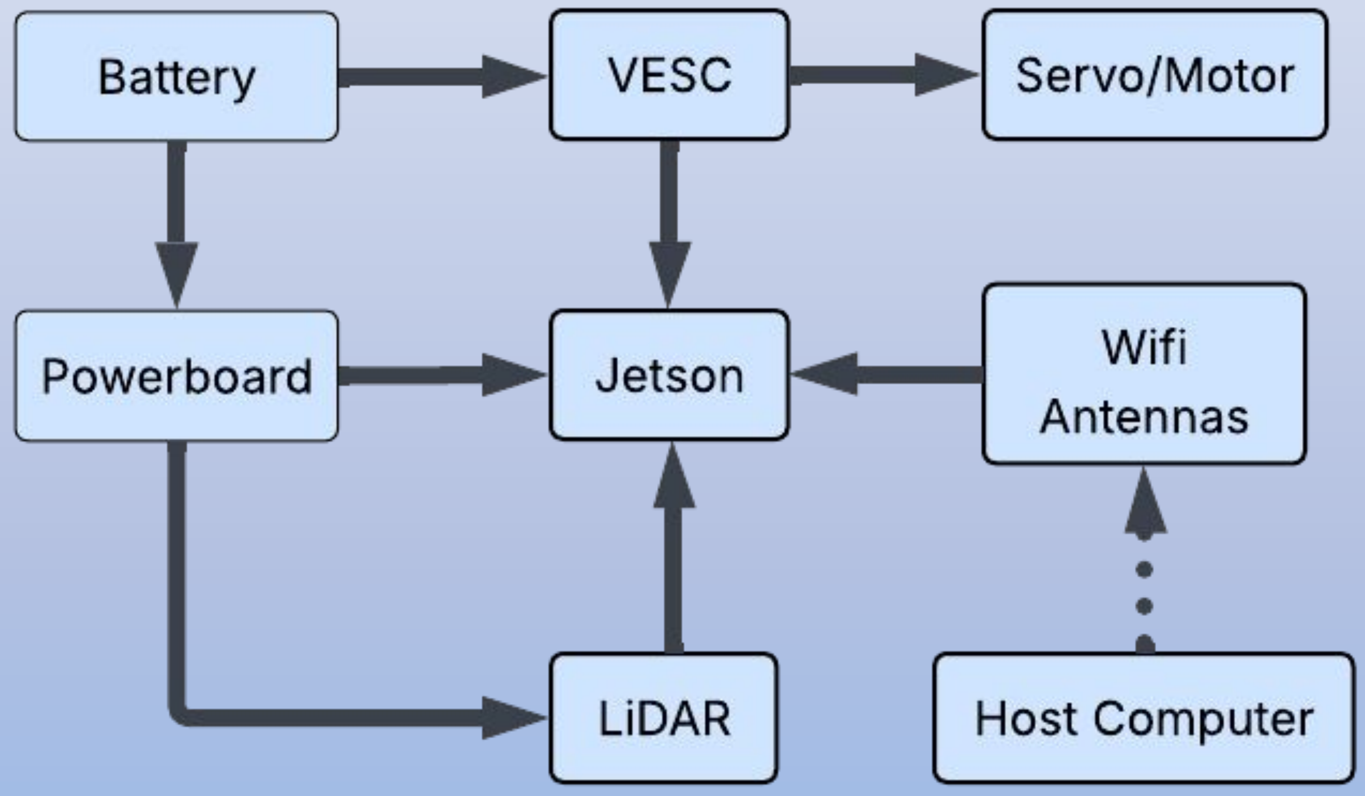
Hokuyo UST-10LX
LiDAR with a 270° scan angle, 40 Hz scan frequency, and a 30 ft detection range



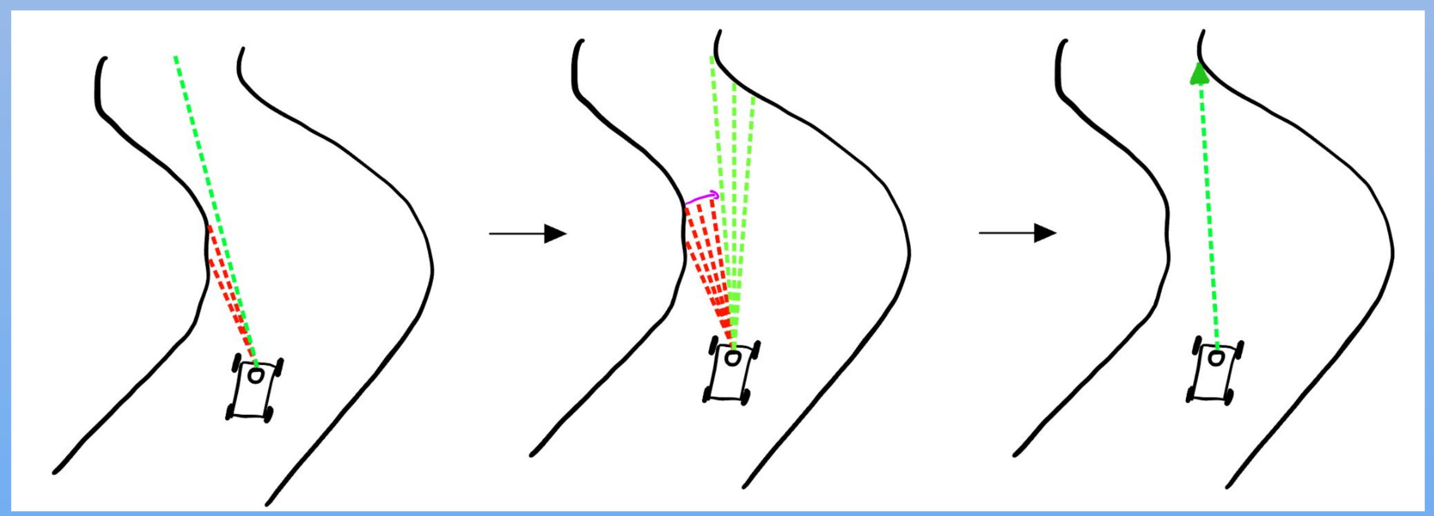
Nvidia Jetson Orin Nano
Computer that reads and processes LiDAR data and controls powertrain

VESC 6 Mk *VI*
Electronic speed controller that receives input from the Jetson and controls the motors

Hardware Block Diagram



Disparity Extender Algorithm



Our car follows a reactive algorithm called disparity extender. First, a disparity, a significant difference between two adjacent distances readings, is found by parsing through the LiDAR data. The disparity is extended, meaning only points from half the width of the car over from the shorter reading of the disparity are considered. Finally, of those points, the farthest one is chosen and the car drives towards it. The speed of the car is linearly scaled based on distances from walls.

Design Intuition

Notable differences between our car and other teams lie in the size of the car and the version of ROS. For this year, the competition had a wide track at 3 - 4 feet wide. As such, we figured that if we selected a smaller car, we would have a higher margin of error on the track. Other teams opted to use ROS2 which is the newer version of ROS. We used ROS1 which made it easier for our packages and dependencies to work right out of the box. Our work culminated in an end of the year competition between the F1Tenth teams on June 2nd, 2025.

Acknowledgements:
We would like to thank Christopher Cheney, Madeline Hesse, Ilan Ben-Yaacov and João Hespanha for their help, guidance and support throughout this project.

robo racer
SIMPLE / FAST / OPEN SOURCE

UC **SANTA BARBARA**
College of Engineering