Race it till you make it

NERVZ

F1Tenth Capstone Project

Navjit Singh
Erk Sampat
Robert Kastaman
Varsha Mattaparti
Zihan Zheng

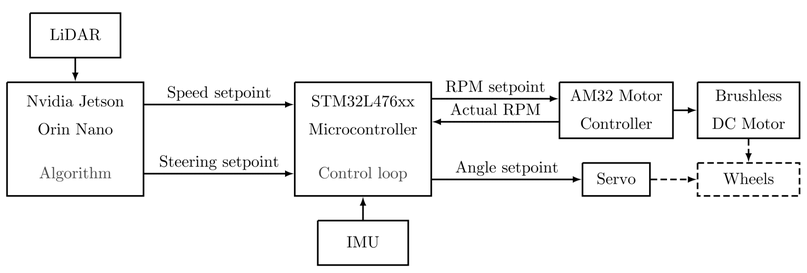
BACKGROUND

We built NERVZ to compete in the F1Tenth Autonomous Racing Competition which is a global engineering challenge where 1/10-scale cars must navigate a multi-lap racecourse using a LiDAR sensor. No human control, no GPS. The race demands split-second decisions: the car must detect walls and obstacles, plan a path, and actuate its motor and steering, all in real-time. To win, you need a fast car, a smart algorithm, and a system robust enough to handle the unexpected. We designed every part of our car from scratch to do exactly that.

OVERVIEW / DESIGN SPECS

- LiDAR-based environment sensing at 40 Hz, processed on a Jetson Orin Nano running ROS2
- Custom STM32 motion controller
- Direct 3S LiPo power delivery to all components, so no conversion stages
- Fully autonomous: no remote control, no GPS. Only onboard sensing and planning
- Custom 2D Python simulator for rapid algorithm development and testing
- Original Free Space Planner (FSP) algorithm for obstacle avoidance
- Weighted Pairs MMSE algorithm for fast, smooth racing on open track

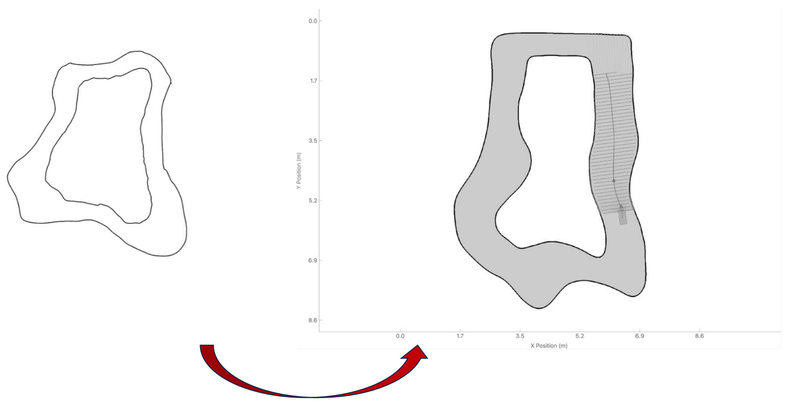
SYSTEM BLOCK DIAGRAM



Above is our high-level block diagram of the entire system. The way this works is simple: The LiDAR scans and records data, the Nvidia Jetson processes that data, and that is going into the STM32 with given setpoints for speed and steering.

From there, the SSTM32 manages the motors, wheels and servo. This is 8 kHz processing that the STM32 performs which is a big highlight of the system.

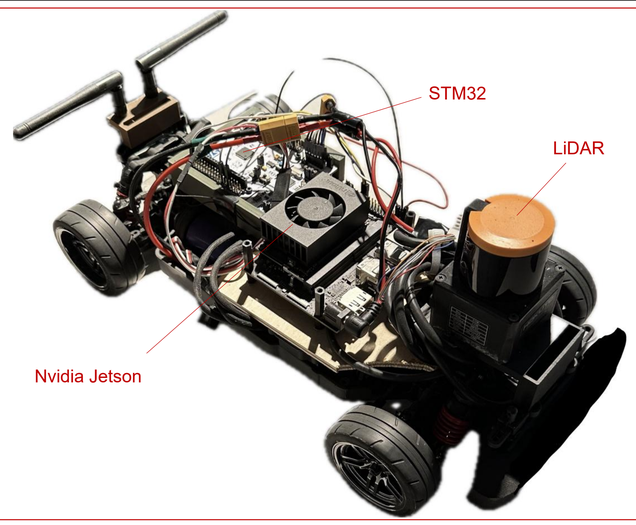
Custom Drawn Tracks



Our custom Python simulator is also used to test a new track: sketch it on an iPad, run it through our image-processing script, and it's loaded into the simulator in under a minute.

We went from using Gazebo, a slow and clunky 3D simulation which resulted in very little progress, to creating a custom 2D simulation which is fast, light, and can take hand drawn tracks made on an iPad and port it over to our simulation for us to test our algorithms on.

FINAL DESIGN



Completed NERVZ autonomous racing vehicle

HARDWARE / KEY COMPONENTS

LiDAR Sensor (Hokuyo UST-10LX)

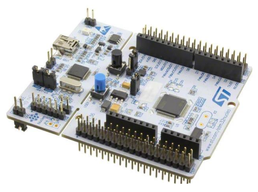
270° field of view, 40 Hz scan rate, ±40 mm accuracy. Connects via Ethernet to the Jetson Orin Nano. Primary source of environment perception which feeds raw distance scans to the ROS2 scan topic.

NVIDIA Jetson Orin Nano

Runs Ubuntu + ROS2. Processes LiDAR data, executes the autonomous planning algorithm, and outputs speed/steering commands at 40 Hz over UART to the STM32 microcontroller.

STM32 Microcontroller

Real-time 8 kHz PID control loop. Drives servo via PWM (steering) and AM32 ESC via DSHOT protocol (motor). Reads RPM via DSHOT telemetry and IMU via SPI for closed-loop feedback.



AM32 ESC

Drone-lineage sensorless BLDC controller. Minimal tuning, small form factor, DSHOT protocol support. Chosen over VESC for its out of the box performance above medium-low speeds.

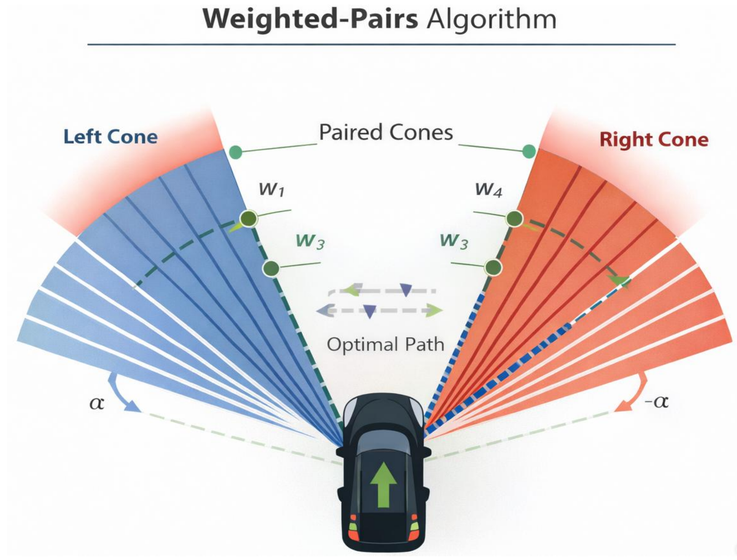
SOFTWARE STACK

- **ROS2:** The LiDAR node publishes scan data to scan, our algorithm node subscribes, computes a drive command, and publishes to drive, which the STM32 UART node reads and executes.
- **Custom 2D Simulator:** We built our own Python simulator. It replicates the car's physical dimensions, turning radius, and speed limits exactly. The LiDAR model is accurate down to adjustable noise and faulty scan simulation. To test a new track: sketch it on an iPad, run it through our image-processing script, and it's loaded into the simulator in under a minute.
- **Dashboard:** Our dashboard visualizes the car's world in real time: the LiDAR rays and their weightings, speed and steering radius commands, average lap time, and algorithm setpoints. It supports simultaneous dual-algorithm comparison and frame-by-frame replay for debugging.

ALGORITHM 1 - WEIGHTED PAIRS

The Weighted Pairs MMSE algorithm is designed for fast, smooth racing on open track sections. It scans the LiDAR's 180° forward field of view and identifies symmetric cone pairs, one left and one right, at the same angle from the car's centerline.

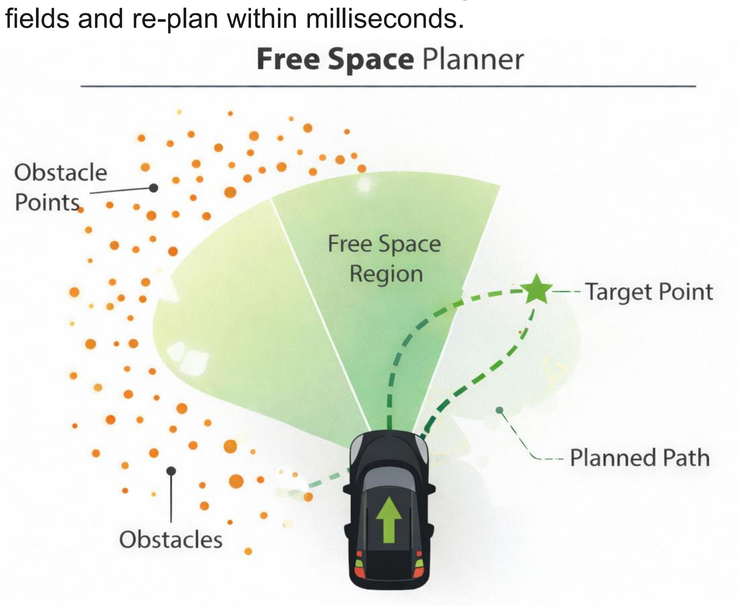
Each cone pair is weighted by distance (closer pairs carry more weight) and an MMSE calculation finds the optimal steering direction that balances all weighted pairs. This produces smooth, centered path-following without building a full map.



ALGORITHM 2 - FREE SPACE PLANNER (FSP)

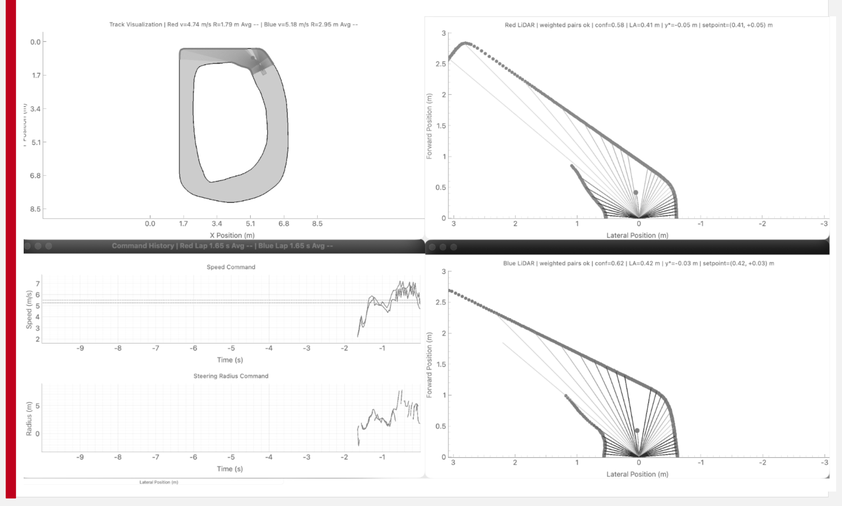
The Free Space Planner (FSP) is a custom A* path planner built on a LiDAR-derived occupancy grid. At each control cycle, raw LiDAR scans are converted into a 2D occupancy grid representing free and occupied space around the vehicle.

A* search finds the shortest safe path to the largest open gap ahead, while respecting tunable safety margins around obstacles. This allows the car to navigate complex obstacle fields and re-plan within milliseconds.



DASHBOARD

Dashboard is simplistic, lightweight, and it shows what the LiDAR sees. We have here Weighted Pairs and FSP side by side.



CONCLUSION

NERVZ is a fully autonomous racing vehicle designed, built, and tested from the ground up over three quarters. Every subsystem, from hardware architecture to the planning algorithm, was engineered ourselves.

Key contributions:

- Custom real-time motion controller (STM32, 8 kHz) for deterministic speed and steering
- AM32 ESC with DSHOT protocol enabling fast, reliable sensorless motor control
- Original Free Space Planner (FSP) using A* on a LiDAR-built occupancy grid
- Weighted Pairs MMSE algorithm for smooth, high-speed racing on open track
- Custom 2D simulator with 1:1 physical accuracy and a track-drawing workflow for rapid iteration
- Dashboard with simultaneous dual-algorithm comparison and frame-by-frame replay

ACKNOWLEDGEMENTS

We want to thank our TAs Chris Cheney and Aaditya Prakash Kattekola for their tremendous weekly support with this project, Professor Ilan Ben-Yaacov for leading ECE 188C, and Professor João Hespanha for his guidance on this project. And the URCA program for the funding provided to assist with this research project.

UC SANTA BARBARA
College of Engineering

URCA