

LAMARRacing: Autonomous Racing

Supervisors: Subham Agrawal and Shahram Khorshidi

Why this project: it combines perception, trajectory planning, and real-time control at racing speed.

Learning outcome: implement perception, state estimation, planning, and control on 1:10 cars.

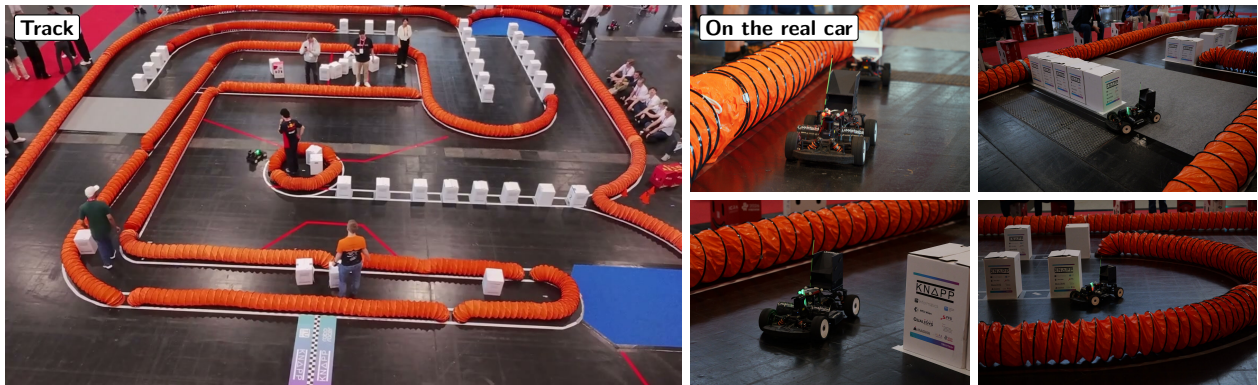


Figure 1: Photos from the ICRA 2026 race: the track (left) and the car racing (right).

Motivation

Autonomous racing pushes a vehicle to its handling limits, where perception, state estimation, planning, and control all have to work together under tight real-time constraints [1]. Scaled cars make this possible on real hardware at low cost and risk, and full head-to-head racing has been shown on fully commercial off-the-shelf hardware [2].

Problem

Implement modules for the race stack that enable autonomous racing. The modules are listed below; a state machine combines their outputs and selects the race and behavior state.

- **State Estimation:** localization of the car on the track, and crash detection.
- **Perception:** detection and tracking of obstacles.
- **Planning:** a local planner and a raceline manager that provide the trajectory to follow.
- **Controller:** turns the plan into commands for the motor controller and steering servo, with a main and a fallback controller.

Goal

Develop modules, integrate them into the race stack, validate them in simulation, and race with them autonomously on the real car.

Task Setting

Up to two groups of two to three students. The groups get a car built on a 1:10 Traxxas chassis with LiDAR/camera sensing and a NUC as the onboard computer. The hardware is available and set up to work. On the software side, the groups get a tailored version of the race stack and the `f1tenth_gym` simulator [3].

Project Schedule

Period	Planned Milestone
Oct 26 to Dec 4, 2026	Crash course (6 weeks).
Dec 7, 2026	Hardware handover.
TBD	Midterm presentation: progress report.
TBD	Final race day and final presentation.

Outcome Levels

Minimum Outcome

- Clean autonomous laps for five minutes.

Expected Outcome

- Racing against opponents with dynamic obstacle avoidance.

Risk and Contingency

The cars operate at high speed and the hardware is fragile. The main risks are damage to the car, scratches on the LiDAR, damaged cameras, and short circuits on the NUC. Damaged hardware must be repaired before further use, and the group has no access to the hardware until the repair is complete. Groups are therefore expected to handle the hardware with care.

Prerequisites

Python is a must. ROS 2 is a plus. Knowledge of control, localization, or perception is very helpful.

Deliverables

A technical report, a demonstration with the real cars racing, and the midterm and final presentations. Both presentations are mandatory.

References

- [1] J. Betz, H. Zheng, A. Liniger, U. Rosolia, P. Karle, M. Behl, V. Krovi, and R. Mangharam, “Autonomous vehicles on the edge: A survey on autonomous vehicle racing,” *IEEE Open Journal of Intelligent Transportation Systems*, vol. 3, pp. 458–488, 2022.
- [2] N. Baumann, E. Ghignone, J. Kühne, *et al.*, “ForzaETH race stack: Scaled autonomous head-to-head racing on fully commercial off-the-shelf hardware,” *Journal of Field Robotics*, vol. 42, no. 4, pp. 1037–1079, 2025.
- [3] M. O’Kelly, H. Zheng, D. Karthik, and R. Mangharam, “F1TENTH: An open-source evaluation environment for continuous control and reinforcement learning,” in *Proceedings of the NeurIPS 2019 Competition and Demonstration Track*, PMLR, vol. 123, 2020, pp. 77–89.