Autonomous driving: an intro

Real-Time Embedded System - The F1tenth autonomous racing

UNIMORE UNIVERSITÀ DEGLI STUDI DI MODENA E REGGIO EMILIA High Performance Real Time Lab

Course outline

- Intro course + basics of AD
- ROS2: Installation and profiling
  - Ex: ROS2 to HiL, open a bag
- Navigation: FTG, FTW, Pure pursuit
  - EX: navigation HiL
- Perception: scan matching, PF, LIO?
  - Ex: perception (PF with PThreads)
- Build the car

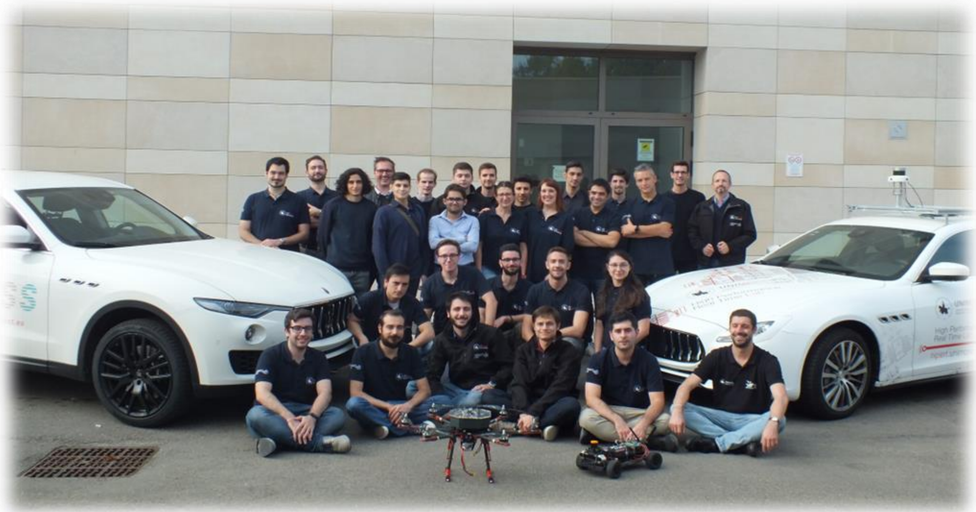
I do <u>not</u> cover all aspects of AD!!!

- Systems and control theory => Prof. Falcone
- Platforms and algorithms for autonomous systems => Prof. Sanudo & Prof. Falcone
- High-Performance Computing => Prof. Marongiu (FIM)
- Machine Learning => Cucchiara's

Credits

- The official “full” F1/10 course
  - Uni Penn/Uni Virginia

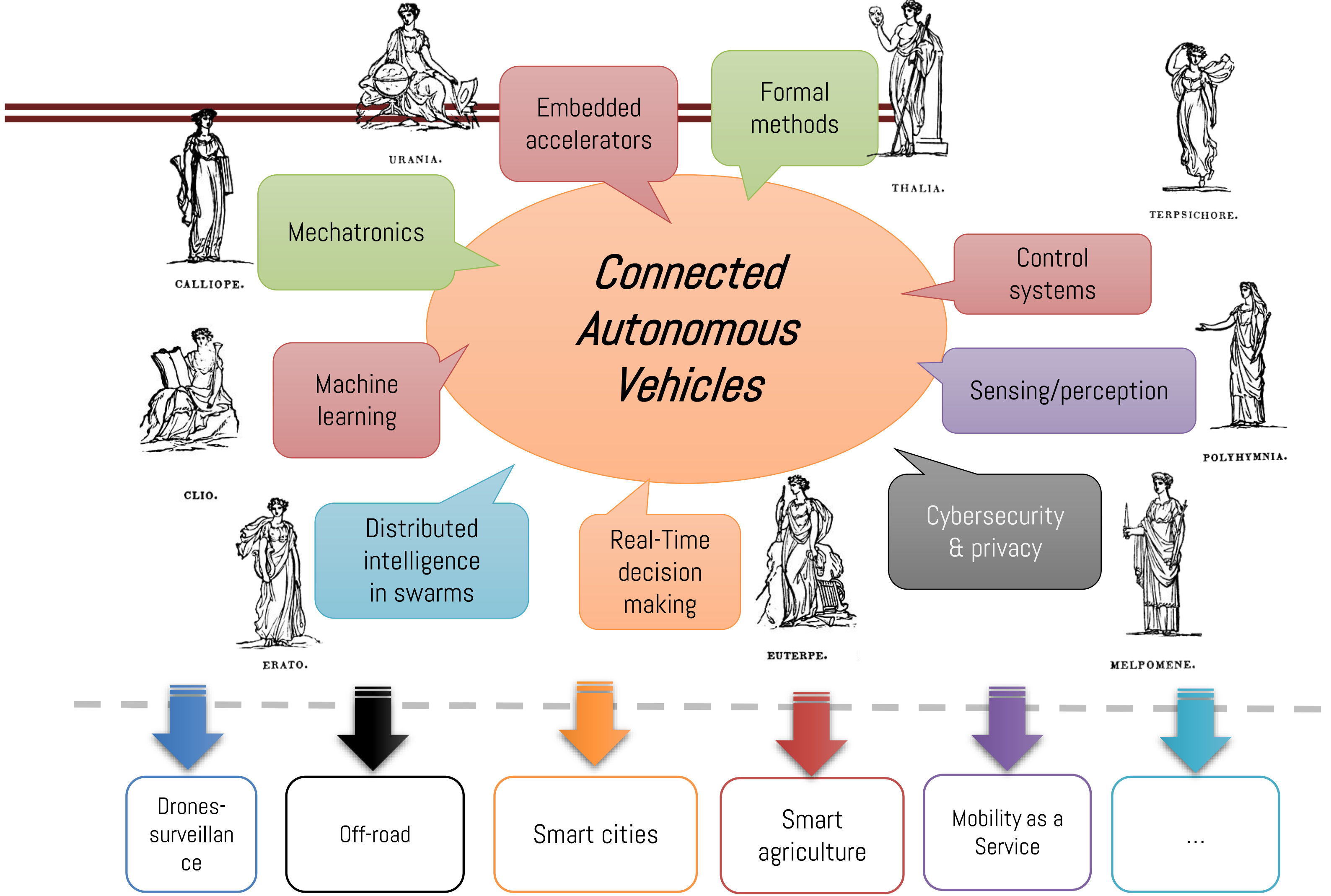
- It's a team work!
  - Hipert Lab & Hipert srl



UNIMORE
UNIVERSITÀ DEGLI STUDI DI MODENA E REGGIO EMILIA

High Performance Real Time Lab

HIPERT



URANIA.
Embedded accelerators
Formal methods
THALIA.
TERPSICHORE.
CALLIOPE.
Mechatronics
Connected Autonomous Vehicles
Control systems
Machine learning
Sensing/perception
POLYHYMNIA.
CLIO.
Distributed intelligence in swarms
Real-Time decision making
Cybersecurity & privacy
ERATO.
EUTERPE.
MELPOMENE.
Drones-surveillance
Off-road
Smart cities
Smart agriculture
Mobility as a Service
...

Self driving cars

Capable of **sensing** the environment

and **moving** safely

with **little or no human input**

- It's hard to model/interpret the world and be able to act intelligently
- Easy to detect the color of a traffic light, not that easy if the sun is directly behind the traffic light

Moravec's paradox: "it is comparatively easy to make computers exhibit adult level performance on intelligence tests or playing checkers, and difficult or impossible to give them the skills of a one-year-old when it comes to perception and mobility"

The importance of standardisation

Level 0

Level 1

Level 2

Level 3

Level 4

Level 5

Level 0
Traditional cars

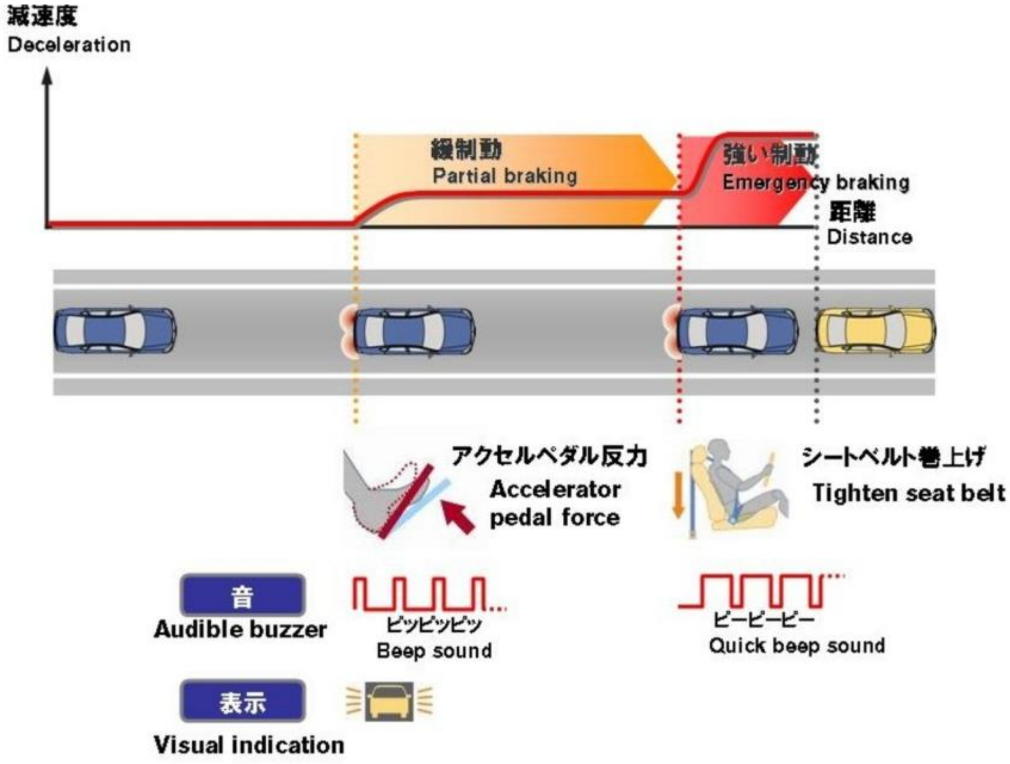
Level 1

Vehicle assists the driver

L1: assistant braking

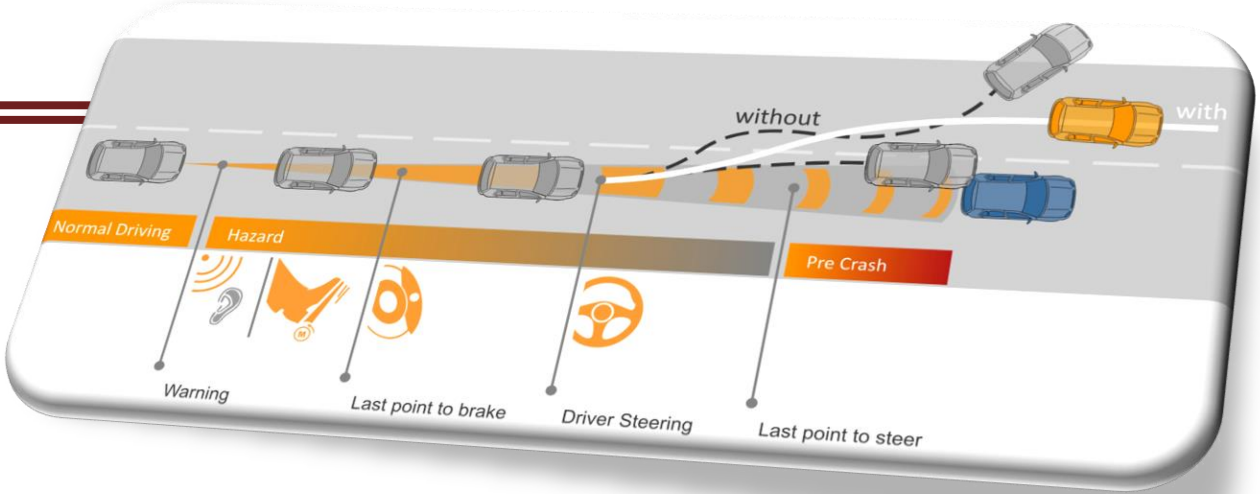
- Nissan's

- Also, warnings
  - E.g., buzzer



L1: lane centering

- Detect lane
  - Does not overtake



Level 2

2+ features, but driver still controls the car

Level 3

Car becomes a co-pilot

Level 4

Car becomes the pilot

Safety-critical driving functions are performed by the car in well-known Operational Design Domains

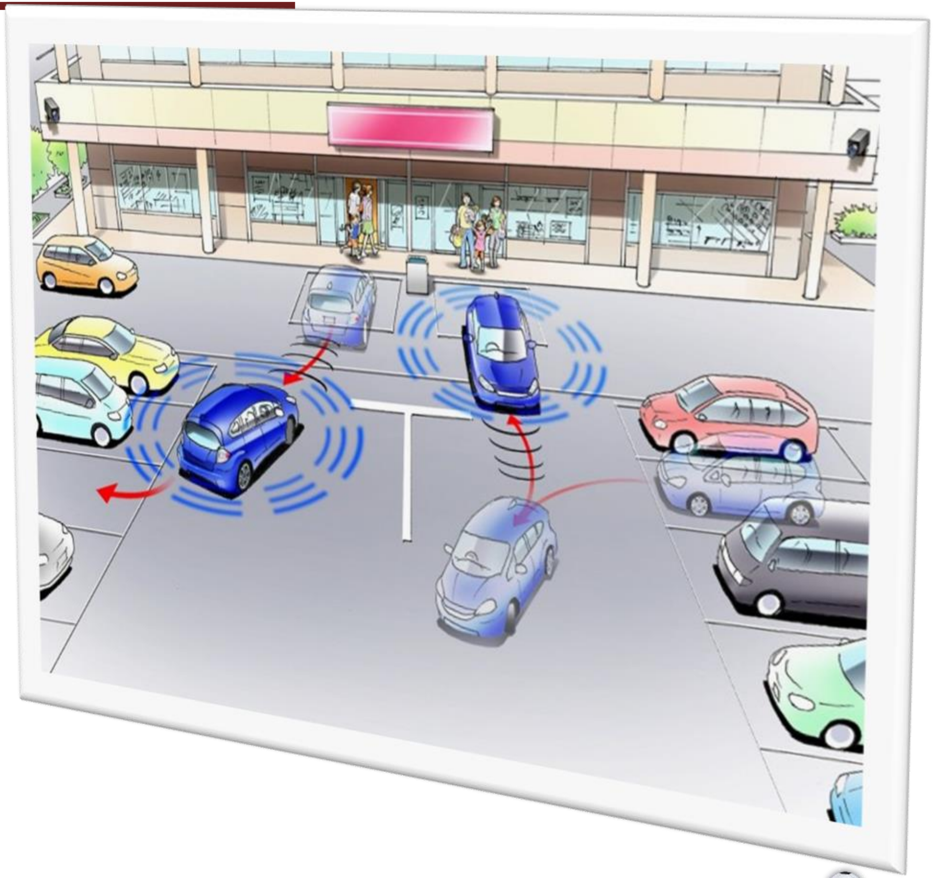
L3 (/4): valet parking

Autonomous search, park and return functionality

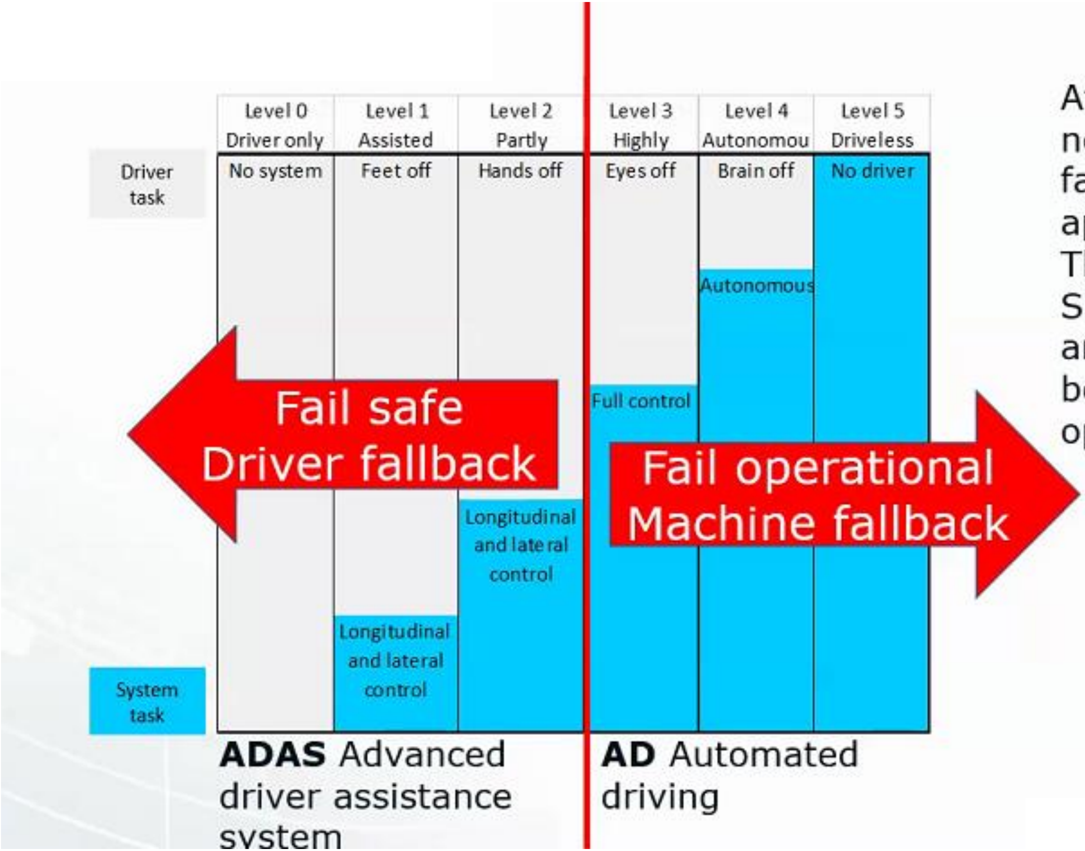
- E.g., at shopping mall, cinemas, ...

No humans, nor cars inside the parking area (ODD)

- Leave your car at entrance

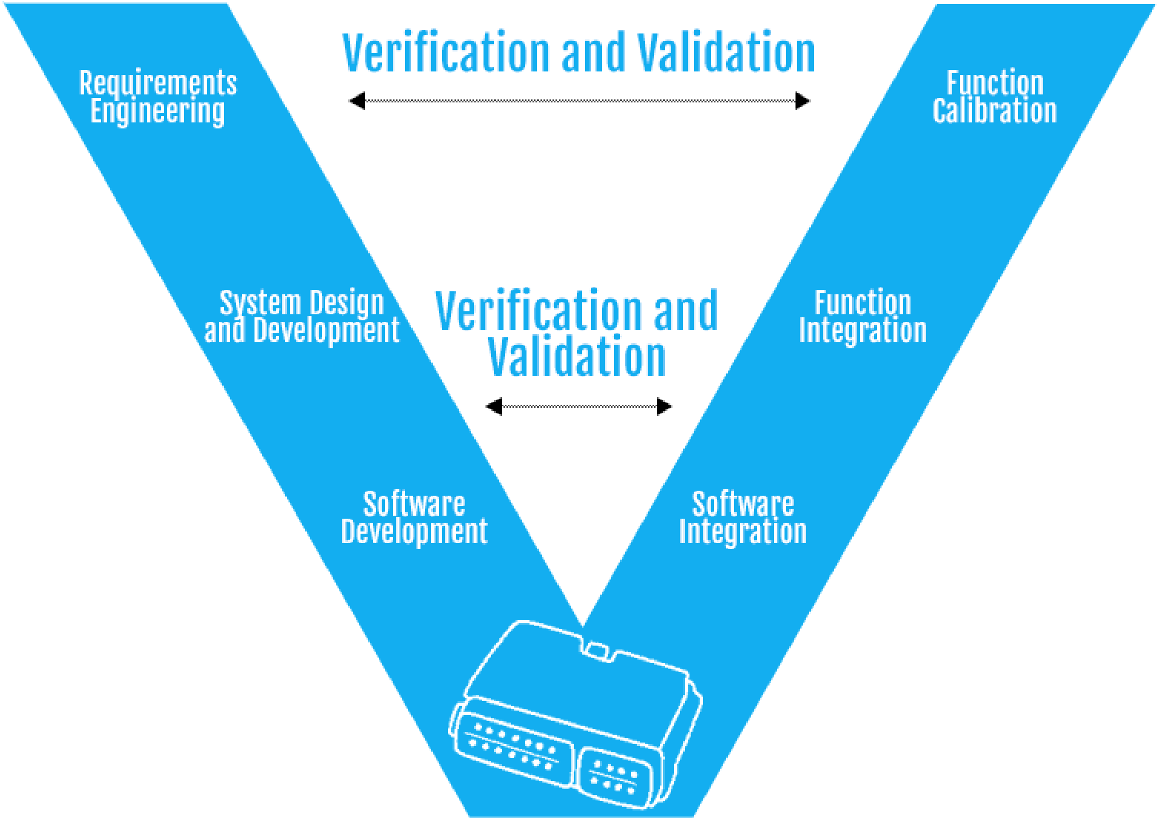


From fail-safe to fail-operational



After SAE level 3 it is necessary to have a fail-operational approach.
The whole system: Sensors, ADAS ECU and actuators, has to be designed fail-operational.

The V-Cycle automotive Software Development Process:

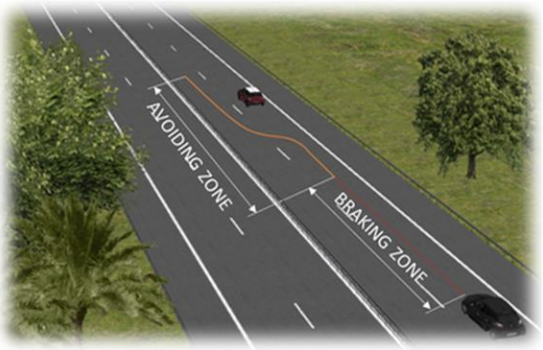


Extraurban Obstacle Avoidance [Danisi, Marelli, 4SG, I&M, POLITO]

NEWCONTROL

A fail operational Extraurban Obstacle Avoidance application to be demonstrated on a real vehicle (UNIMORE Maserati) and on a simulated environment (DANISI dynamic simulator)

FO Aspect: Fail-operational onboard system based on two redundant computers



The goal

› Being able to switch between the two computers during a L4/L5 manoeuvre

The scenario

› Start the manoeuvre
› Inject/simulate a fault in the board 'A'
› The board 'B' seamlessly takes over, and completes the manoeuvre

Use case: Extraurban Obstacle Avoidance [DANISI, Marelli, 4SG, I&M, POLITO, UNIMORE]

NEWCONTROL

Design methodologies	Embedded platform	Algorithms	Virtual Platforms	Demonstrators (**SC6**)

ISO 26262 (Functional safety) methodology for the design of fail-operational automotive board [**Marelli & 4SG**]

SC2
Fail-operational onboard processor
[**I&M**]

SC3
Path Planner & Control
[**PoliTO, UNIMORE**]

SC3
Simulation environment for validation of fail-operational capabilities
[**DANISI**]

Simulation environment for evaluation of super-human performance [**DANISI**]

Maserati Quattroporte
[**UNIMORE**]

Balancing Performance and Safety

Current AV technology still struggles in non-cooperative scenarios like merging due to competing objectives:

- **Maximize performance:** negotiate the merge without delay or hesitation

- **Maintain safety:** avoid catastrophic failures and crashes

Racing (autonomously) highlights this performance safety tradeoff.

Videos: Mobileye and Formula 1

Autonomous Driving: Why racing ?

1

Detecting the vehicle limits

- Unstructured Environment
- Different Tracks and Conditions
- Different vehicle setups

2

Decision making at the vehicle limits

- High prediction uncertainty
- Strategy, Energy, Overtaking

3

Handling at the vehicle limits

- High speeds, High accelerations
- High planning horizon necessary
- Small reaction times

F1 TENTH

Autonomous Racing @Hipert/UNIMORE

FITENTH

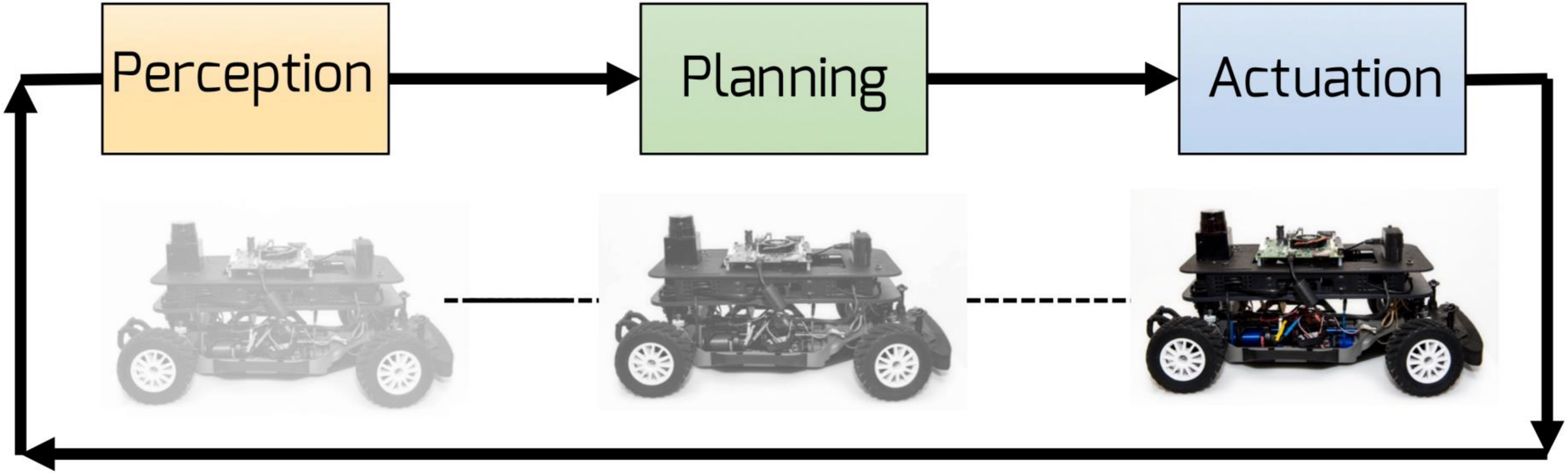
Roborace

Formula Student

Indy Autonomous Challenge

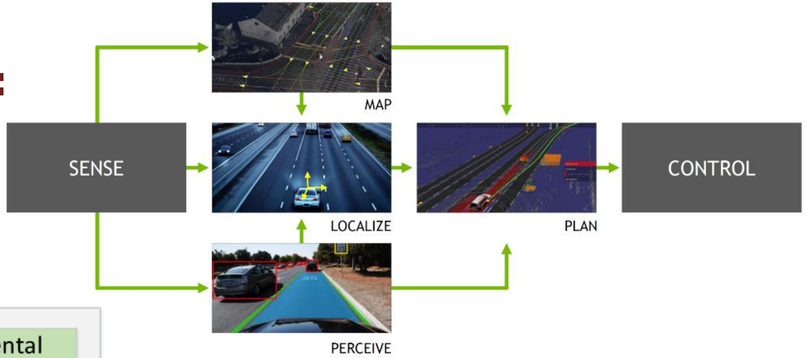
A deeper look....

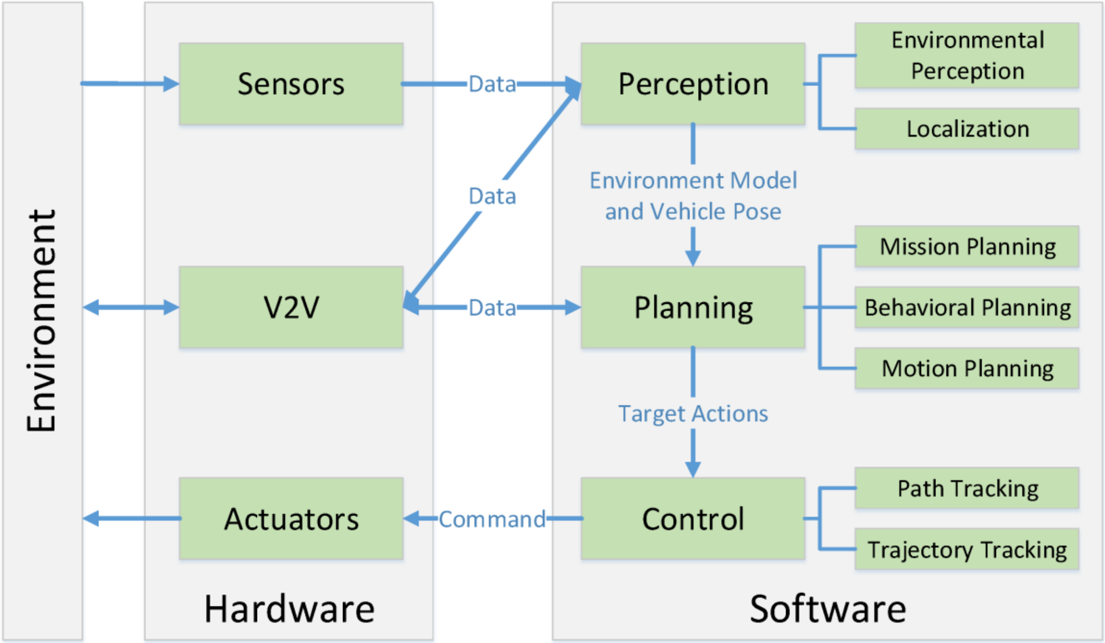
How do we do AD (at UNIMORE)?



(At least) three stages – F1/10

...the tough reality



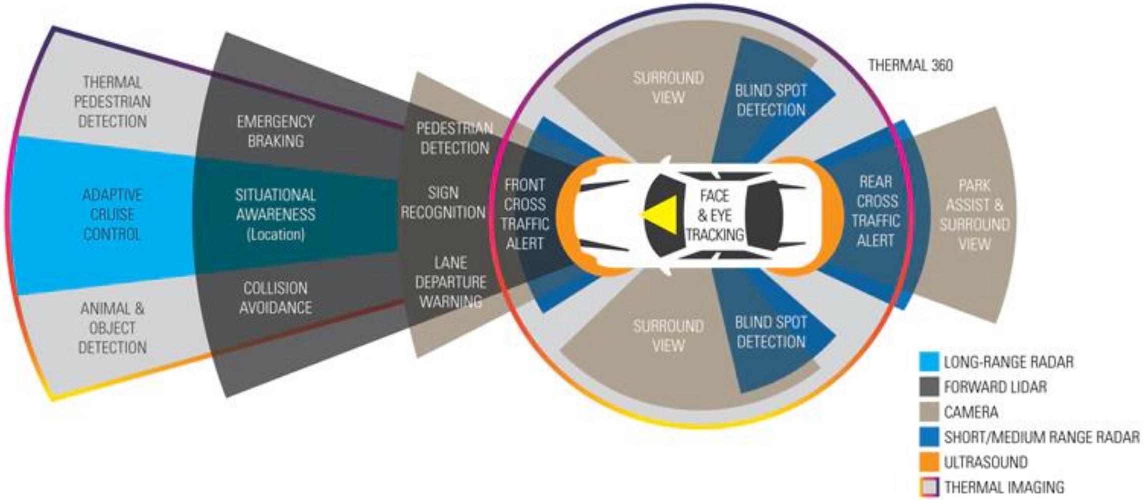


Sensors

› **Exteroceptive**: capture the external environment

› **Propioceptive**: capture of the ego vehicle

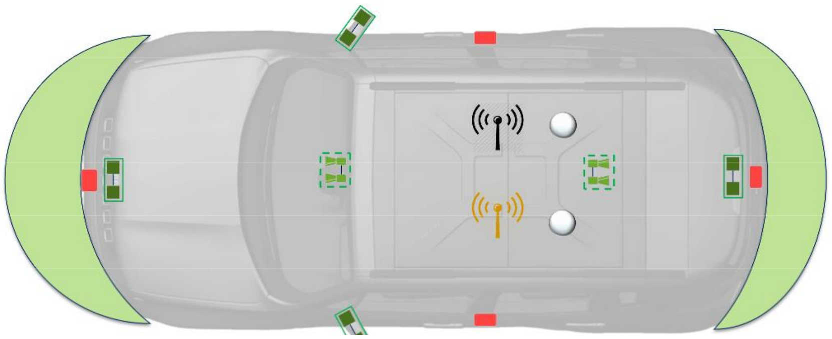
› The number and complexity of the sensors increased dramatically in the last few years



HiPeRT/UNIMORE Car

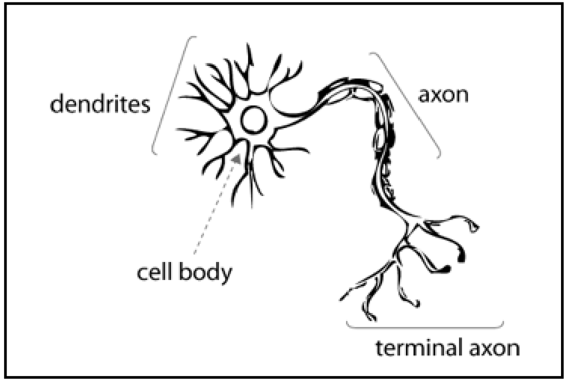
Maserati quattroporte

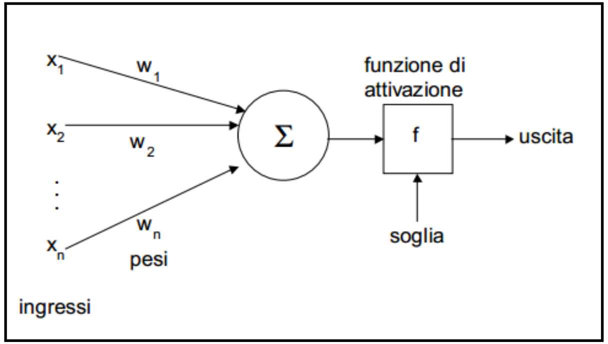
- 5 sekonix cameras (GMSL)
- Ouster OS1 64/128-120m (Ethernet)
- GPS (CAN)
- Radar ARS301
- Multiple ECU (Electronic Control Unit)
  - Pegasus
  - Drive PX2
  - Xavier
  - TX2



Neural networks

- Bio-inspired
- Based on neurons arranged in layers
  - And sub-layers
- Convolutional neural network
  - Perform Convolutions

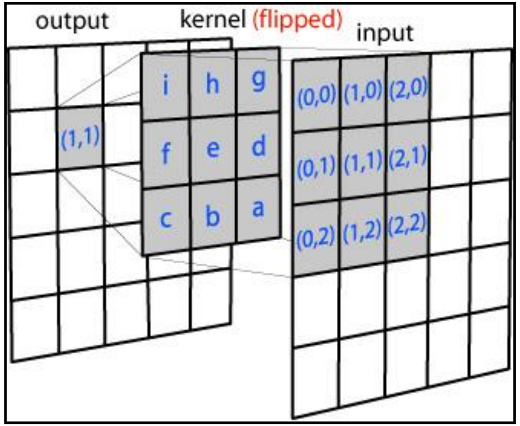




Convolution

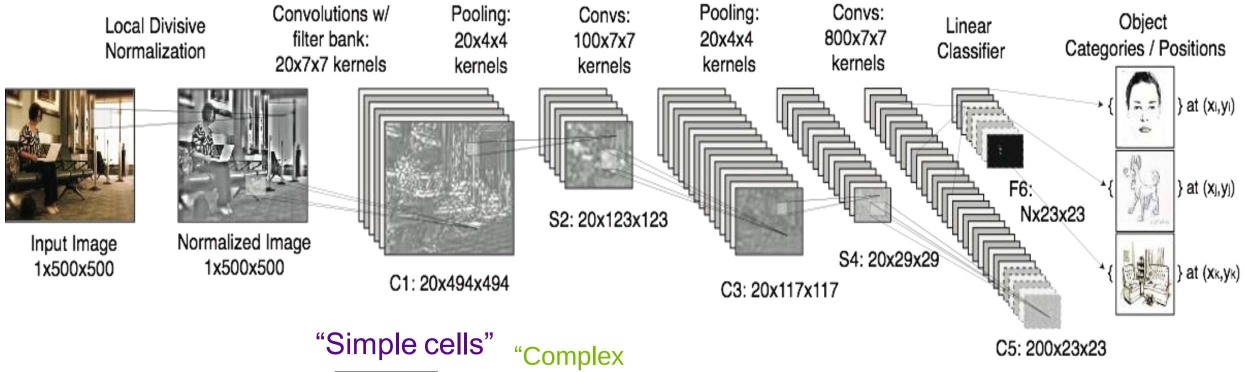
- Computation-intensive
- Suitable for implementation in hardware
- In computer vision, blurring

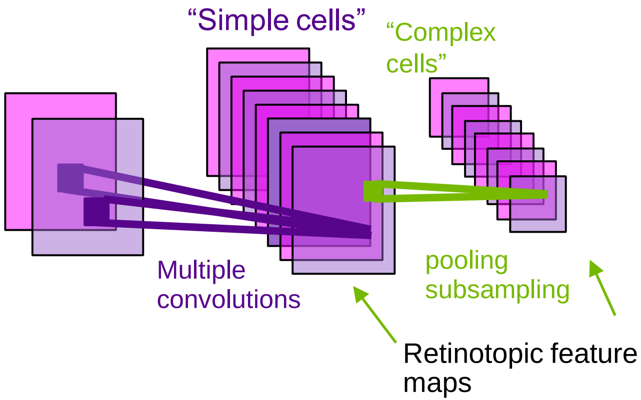
$$
\begin{aligned}
(f * g)(t) &\stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(\tau) g(t-\tau)\, d\tau \\
&= \int_{-\infty}^{\infty} f(t-\tau) g(\tau)\, d\tau.
\end{aligned}
$$



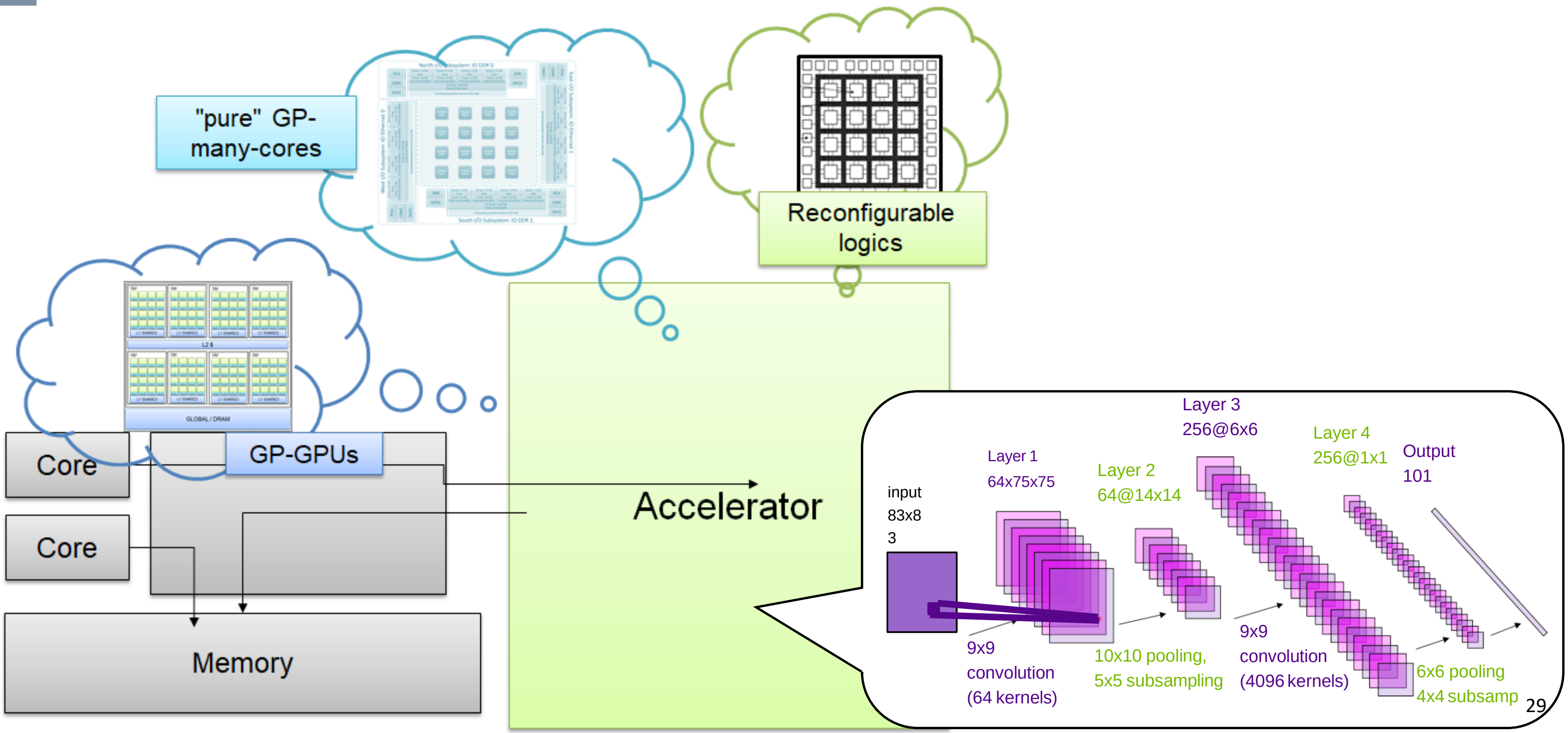
The convolutional net model

- (Multistage Hubel-Wiesel system)



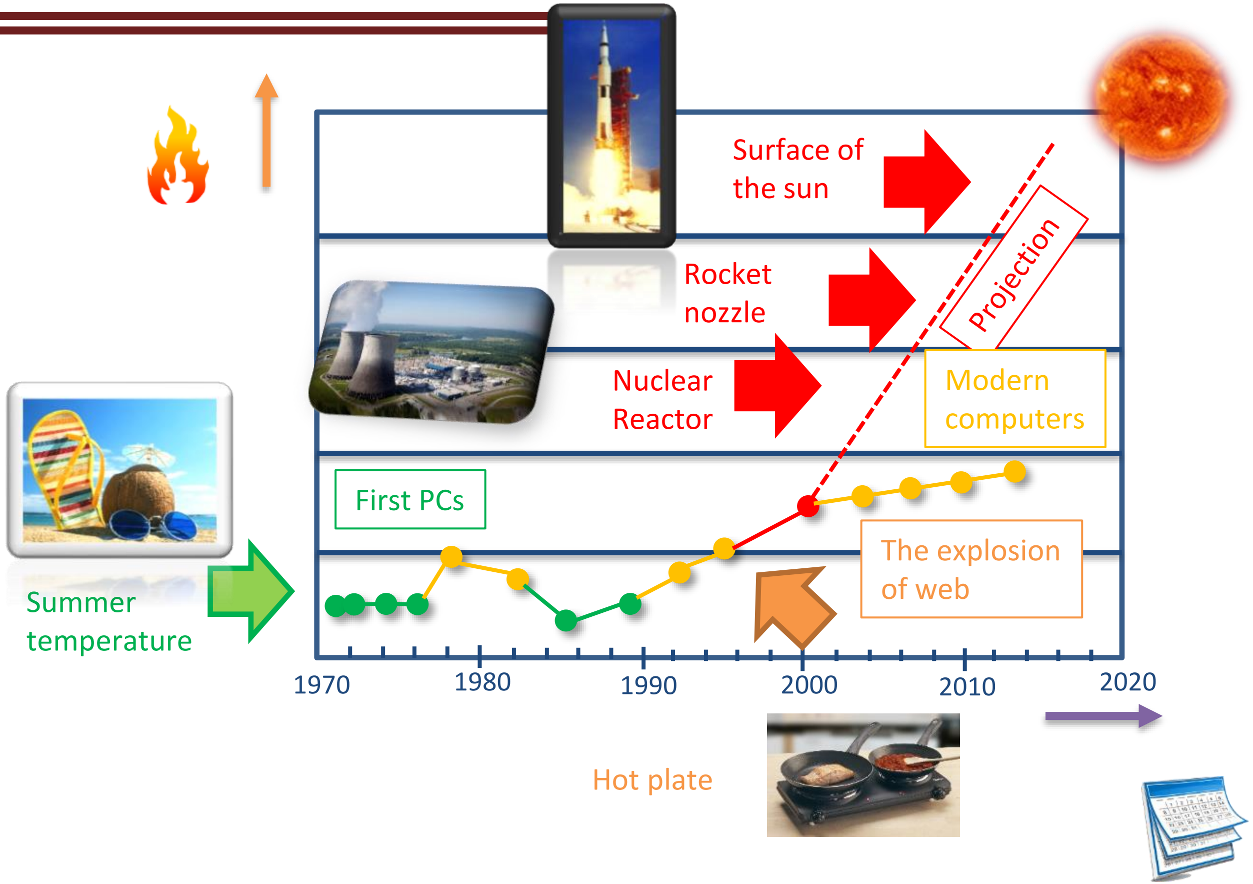


The convolutional net model (cont'd)



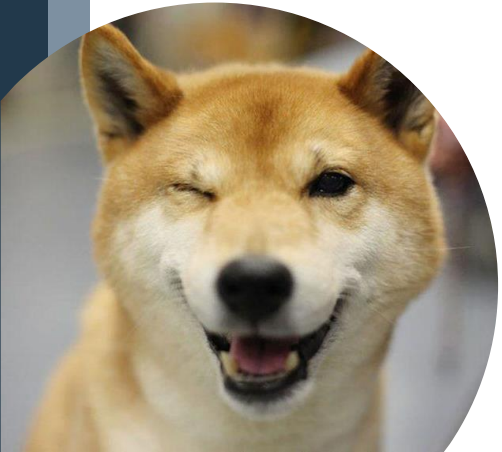
“What's the problem?”

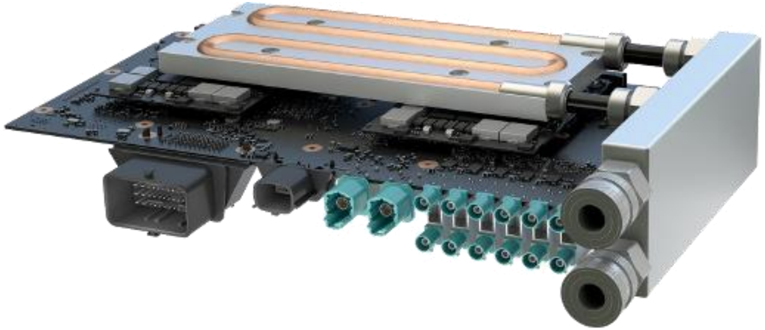
I'm hot, baby!



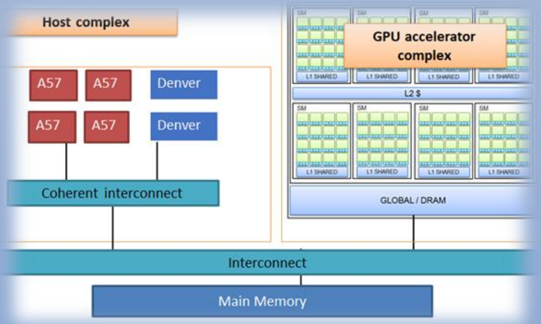
Surface of the sun
Rocket nozzle
Nuclear Reactor
Projection
Modern computers
First PCs
Summer temperature
The explosion of web
1970
1980
1990
2000
2010
2020
Hot plate

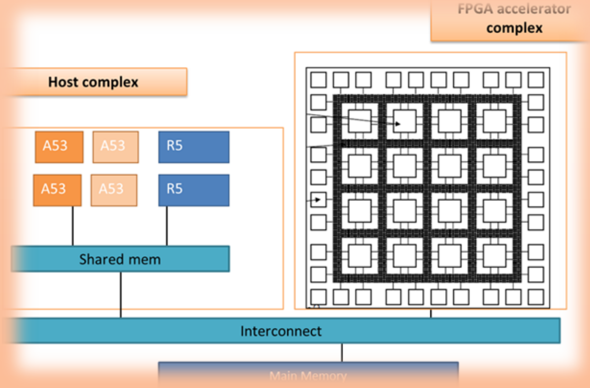
Embedded systems





GPGPU
SKY HARD





W LA FPGA

Real-Time Embedded Systems

Aka: Why we're all here..

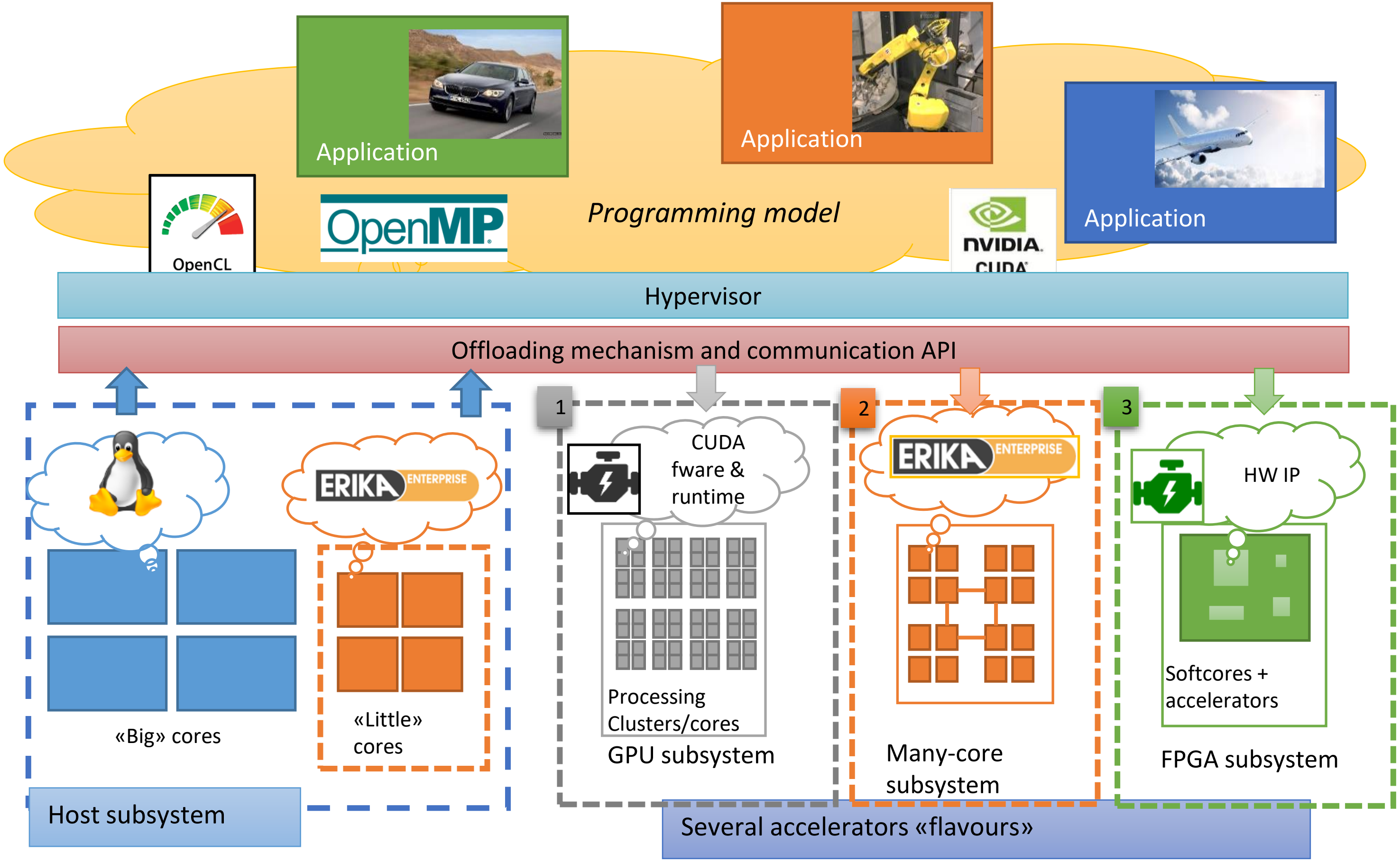
Cars are constrained!

- Size, Weight and Power – SWaP
- High workload (e.g, Machine Learning), require high-performance and power efficiency
- Example: How does it take to turn on MR-23DL ("Diletta")?

Safety critical systems require predictable performance

- At all the level of the software stack
- Not only application...but also RTOSes, hypervisors...

An example: the Hercules stack



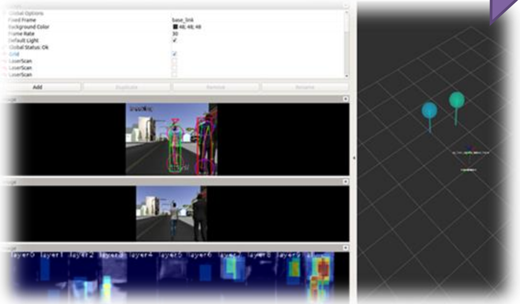
How do I become a professional AD engineer?

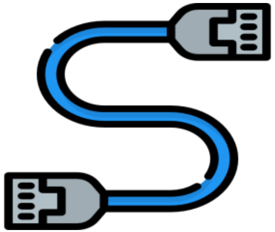
Can I learn it?

First grade - simulators

- GTA-V
- Assetto Corsa
- Carla
- LGVSL
- …

- Python
- C++
- Matlab
- …

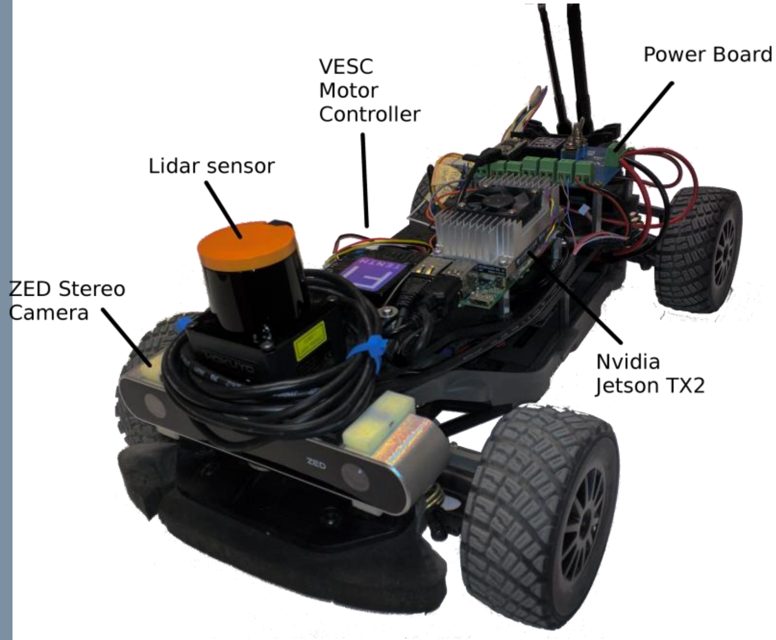




(Optional)
On-board computer

Hardware-in-the-Loop

Second grade - the F1/10 challenge



A scale car

- Instructions are provided
- Challenge is on software
- 2-3 Challenges per year, span across the world

Less than 3k€ to build the car

› Join the race!

Third grade - the real car

For students

- Professors are just advisors; teams are 100% self-managed (even fundings!)

Very good for CV and team/soft skills

- Part of our BS/MS courses

Build a fully autonomous racing car

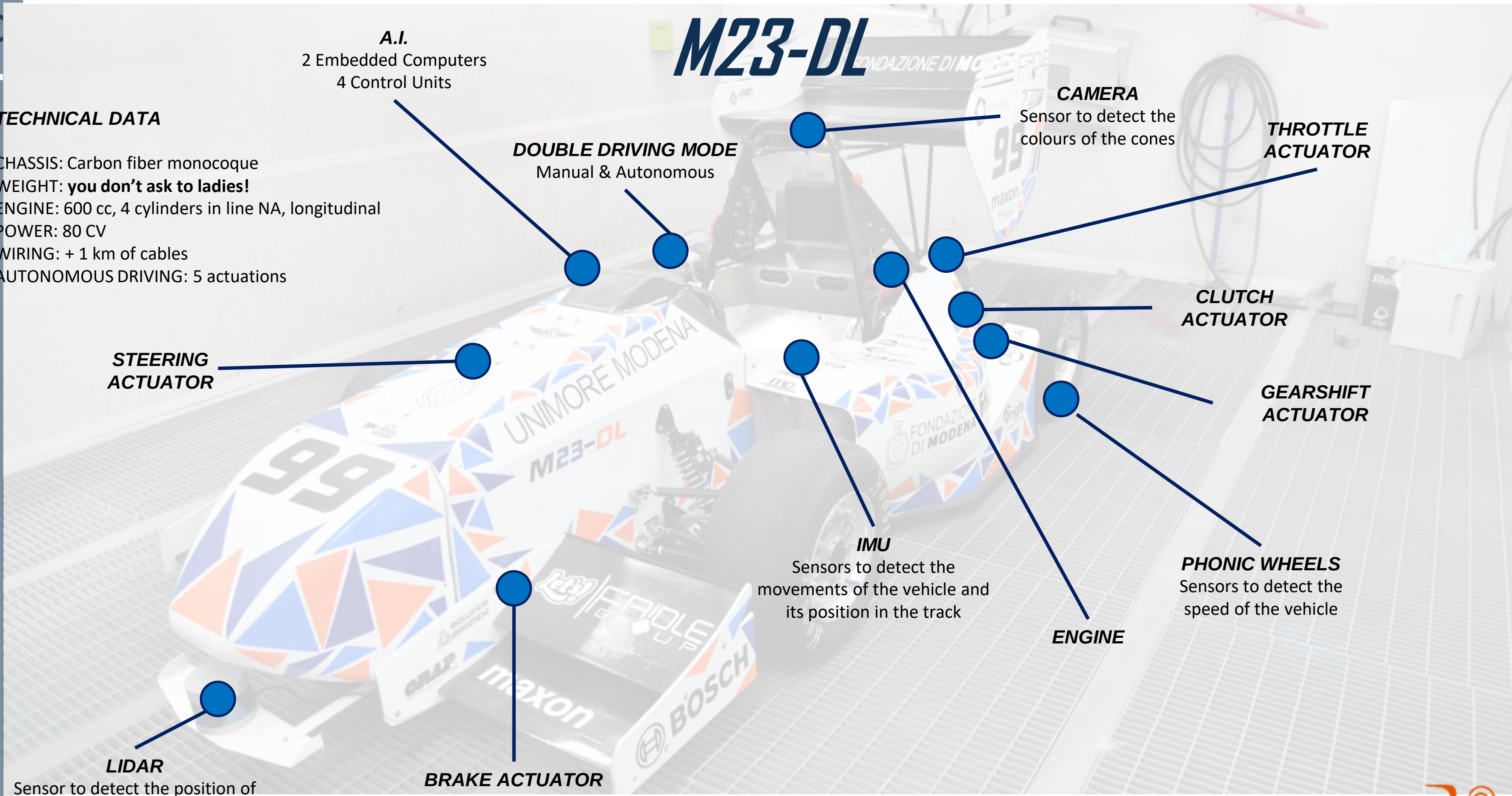
- Not only engineers...a full team, with marketing, management...
- Competition also on project only...no need to race or build the car

I can get ~~no~~ satisfaction

M23-DL

TECHNICAL DATA

CHASSIS: Carbon fiber monocoque
WEIGHT: **you don't ask to ladies!**
ENGINE: 600 cc, 4 cylinders in line NA, longitudinal
POWER: 80 CV
WIRING: + 1 km of cables
AUTONOMOUS DRIVING: 5 actuations

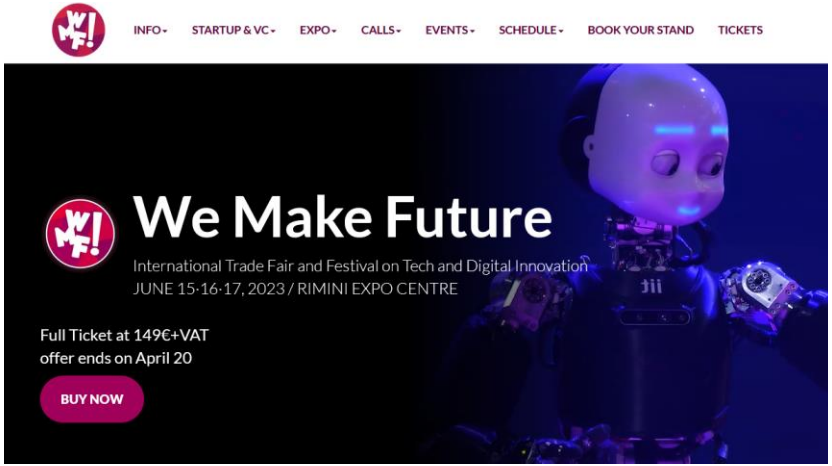


The Team 2021-22

Formula ATA, Varano De' Melegari (PR), July 2022

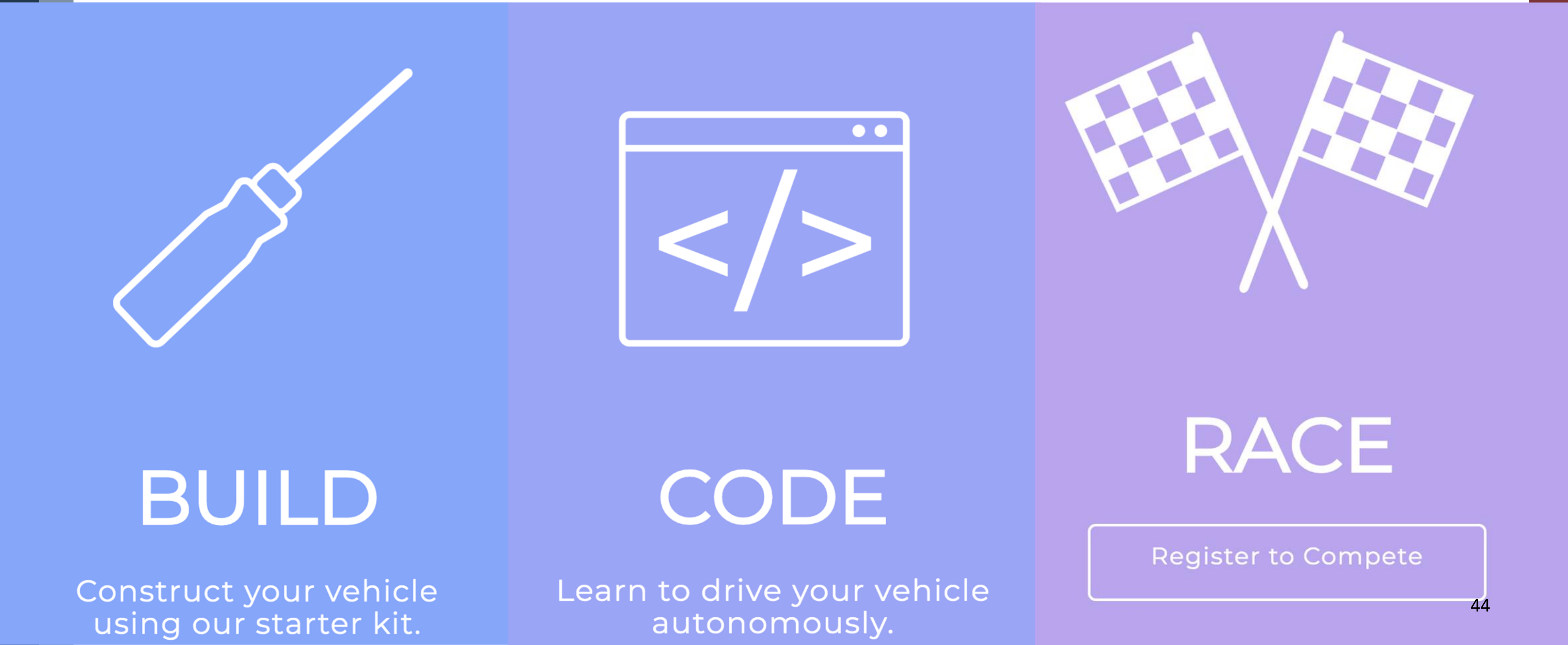
We want you!

- We Make Future fair, UNIMORE vs. UNIBO
  - https://en.wemakefuture.it/
  - 15-16-17 June 2023

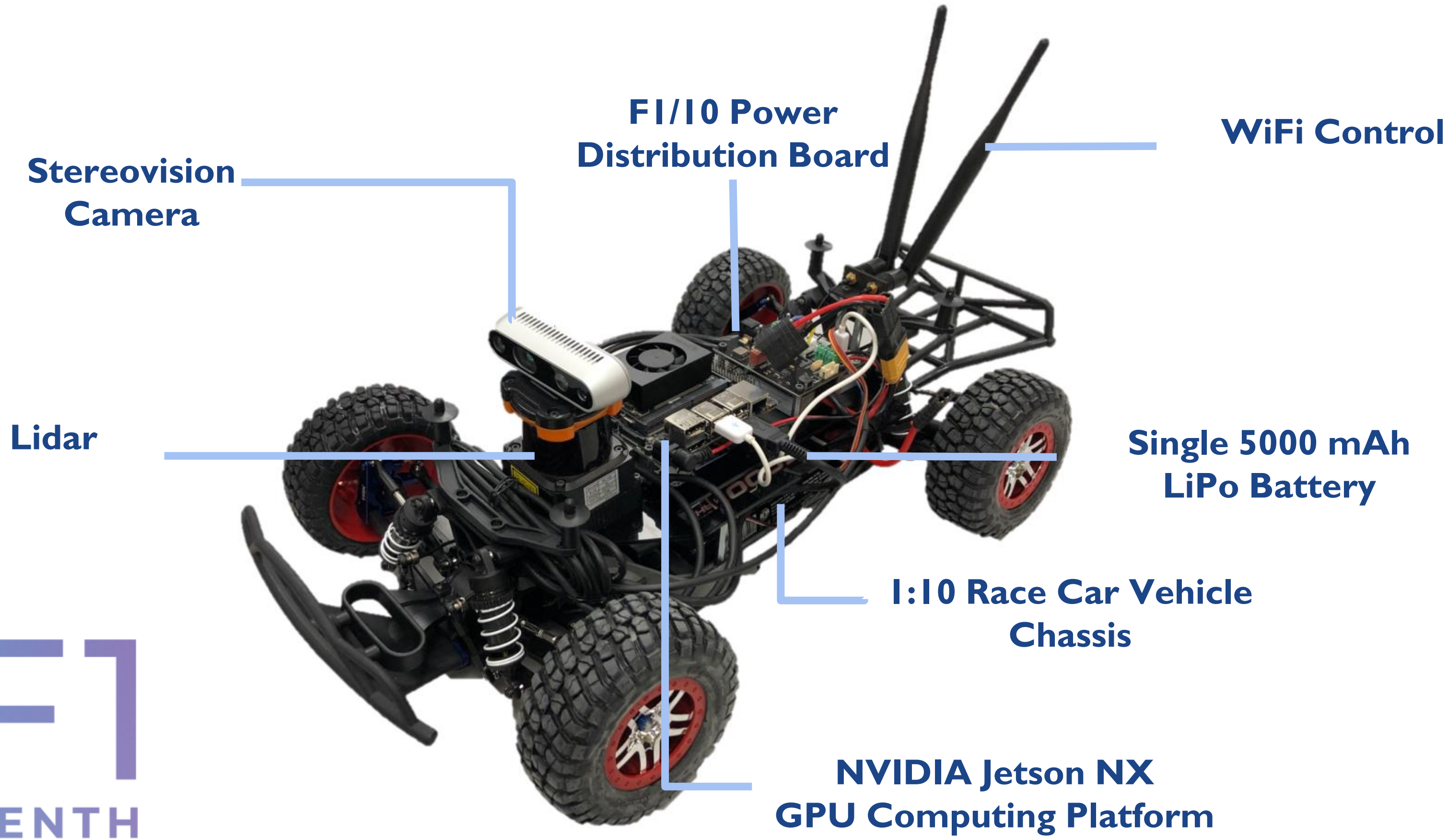


We need 1-2 persons to run the race/show...

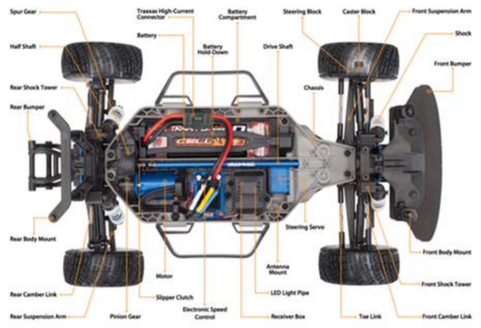
F1/10: 1/10 the size, ~~10x the fun!~~ (1/10 the problems)



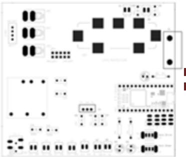
Low-Cost Open-Source Platform



F1 TENTH



Chassis Design

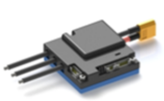


F1/10 Powerboard

2D Lidar

Stereo Camera

Mono Camera



VESC Controller

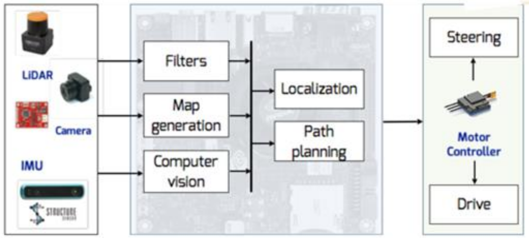
Wifi

NVIDIA Jetson NX GPU Compute Platform

System Integration

Software Architecture

ROS2



Perception Planning Control

GPU Acceleration

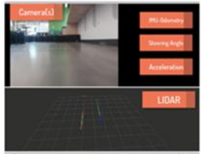
Research Enabled

Safe Autonomy	Secure Autonomy	Coordinated Autonomy	Efficient Autonomy

Lane Keeping Assist

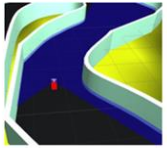
Model-Predictive Control

AV Data Collection



Simultaneous Localization And Mapping (SLAM)

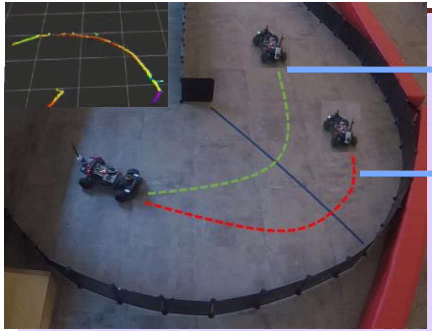
F1/10 Simulator



DNN Racing

F1 TENTH

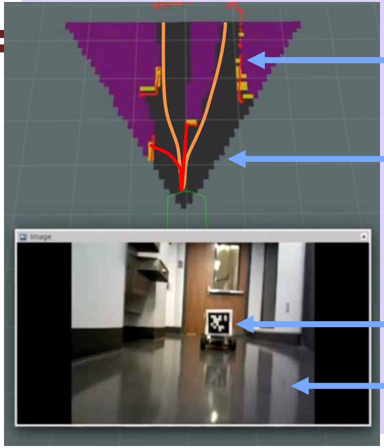
FOLLOW THE GAP METHOD



Follow the gap method

Simple obstacle avoidance

MODEL PREDICTIVE CONTROL



Generated paths

Drivable region

Dynamic obstacle

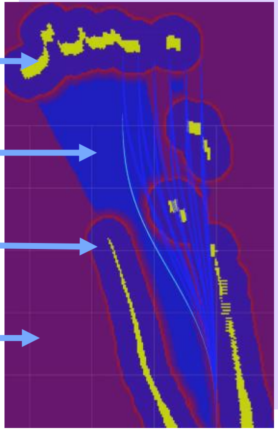
F1/10 RGB FPV

Max. cost (obstacle)

Drivable region

Medium cost

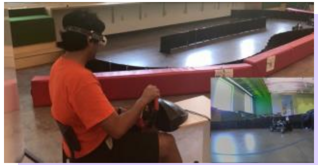
Unexplored space

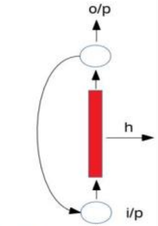


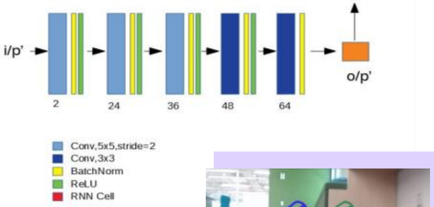
Cost Map Legend

END-TO-END DNN

FPV data annotation



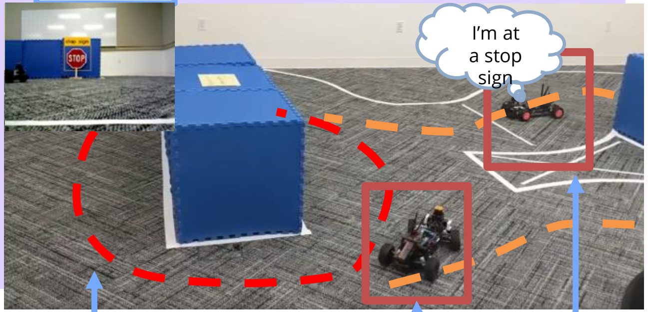




Predicted steering (blue)
Ground Truth (green)

V2V COLLABORATION

F1/10 FPV

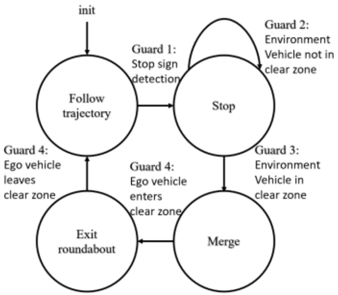


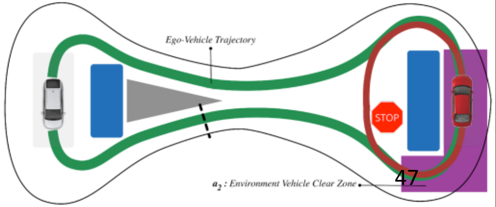
Env. Vehicle Path

Environment vehicle

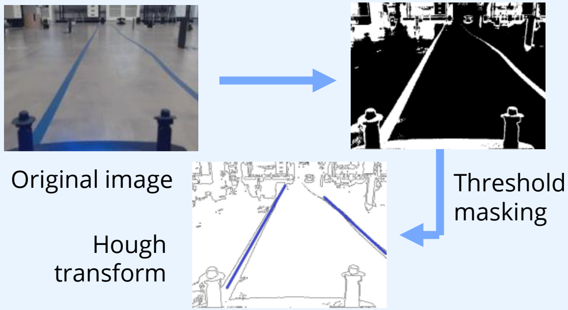
Ego vehicle

Collaborative behavior automaton

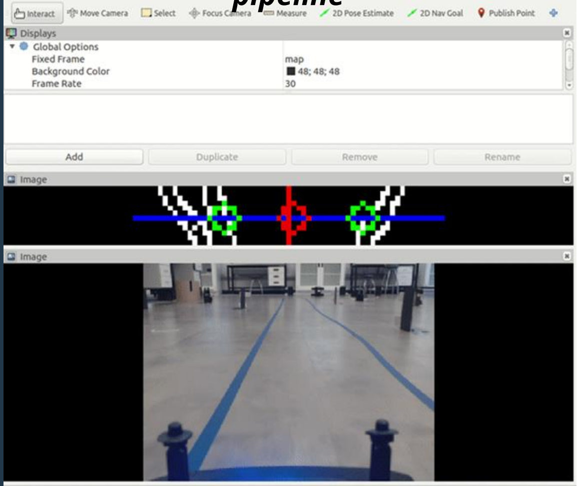




LANE KEEPING ASSIST

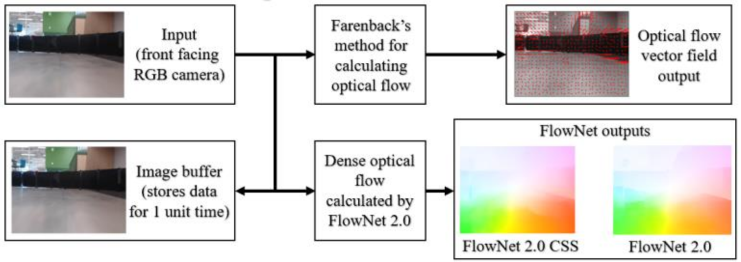


Lane detection pipeline

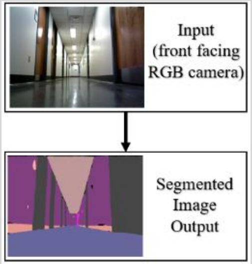


rviz visualization

COMPUTER VISION EXAMPLE



Optical flow



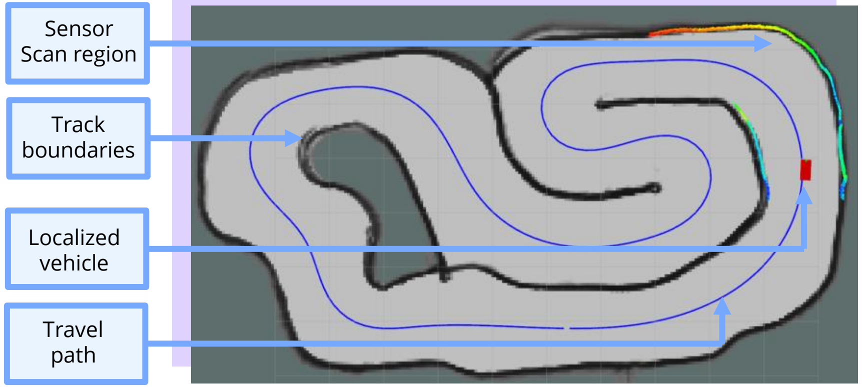
F1/10 online semantic segmentation

F1 TENTH

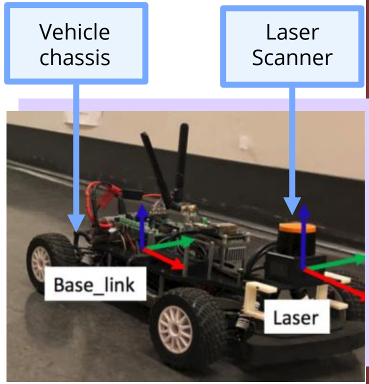
LOCALIZATION AND MAPPING

Currently tested On F1/10

Hector SLAM	AMCL	Google Cartographer	ORB-SLAM



Global planning using *rviz*



ROS transform frame

F1TENTH SIMULATORS

Test algorithm → ROS2 → F1Tenth simulator

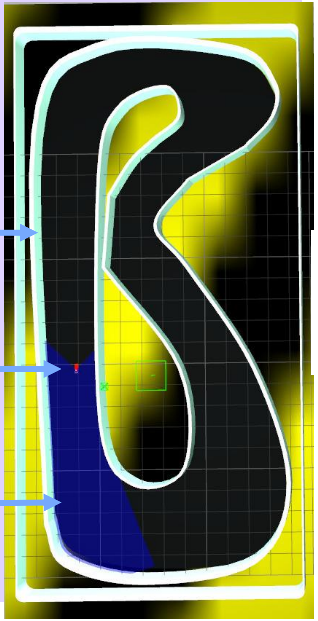
F1/10 ROS URDF model → ROS2

Overhead view of track in Porto

Track boundaries

Simulated F1/10 car

Sensor Scan region

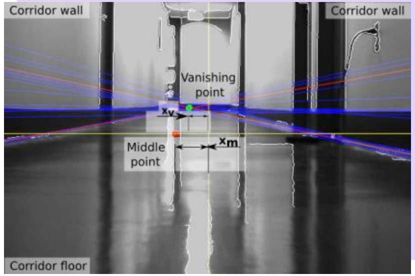


Porto track built using ROSbag data

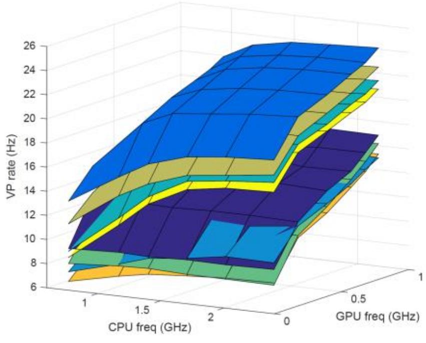
→ ROSbag logs LiDAR and Odometry data →

REAL TIME SCHEDULING

Vanishing point (VP) algorithm implemented on F1/10

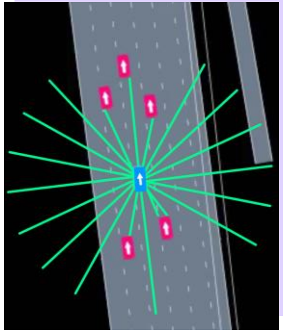


Green dot (vanishing point)
Red dot (middle point)



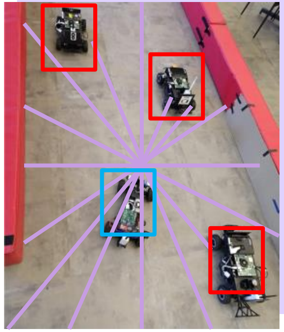
VP throughput based on CPU & GPU frequencies

TESTING & VERIFICATION



Data captured from traffic on I-80

GAIL model: Outputs include mean & variance of steering & throttle



F1/10 helps recreate traffic scenarios to verify model

Lesson(s) Plan

Part I: Learn to Drive

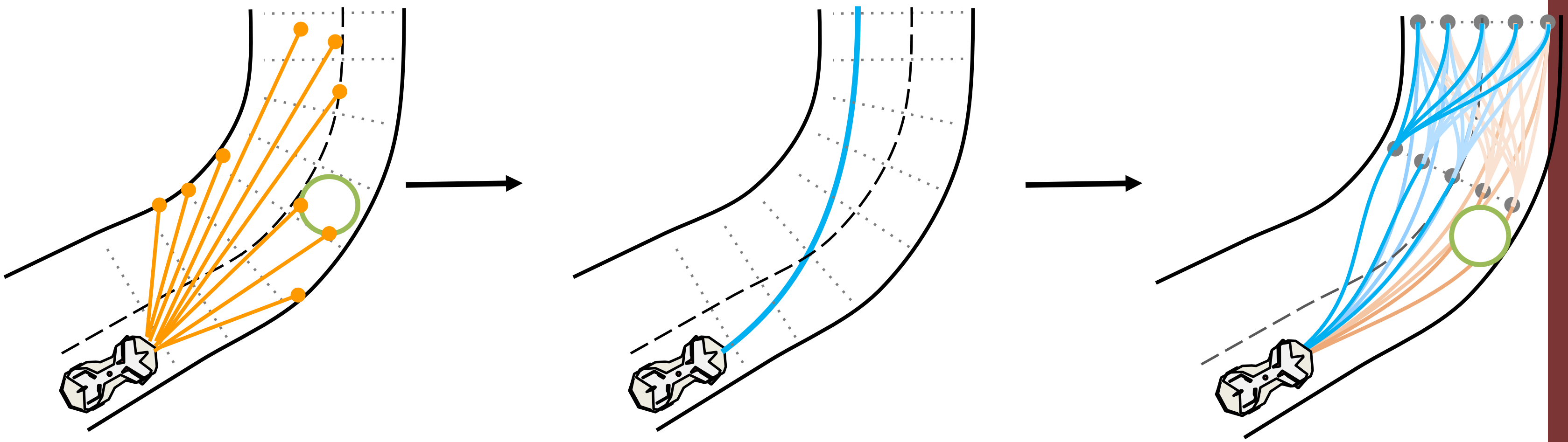
Autonomous Driving w/ F1TENTH

Avoid the crash - Reactive Planning

Part II: Learn to Race

Follow the Raceline - Pure Pursuit

Race & Overtake - Graph Planner



1. Follow The Gap **2. Follow The Raceline** **3. Race**

List of assignments

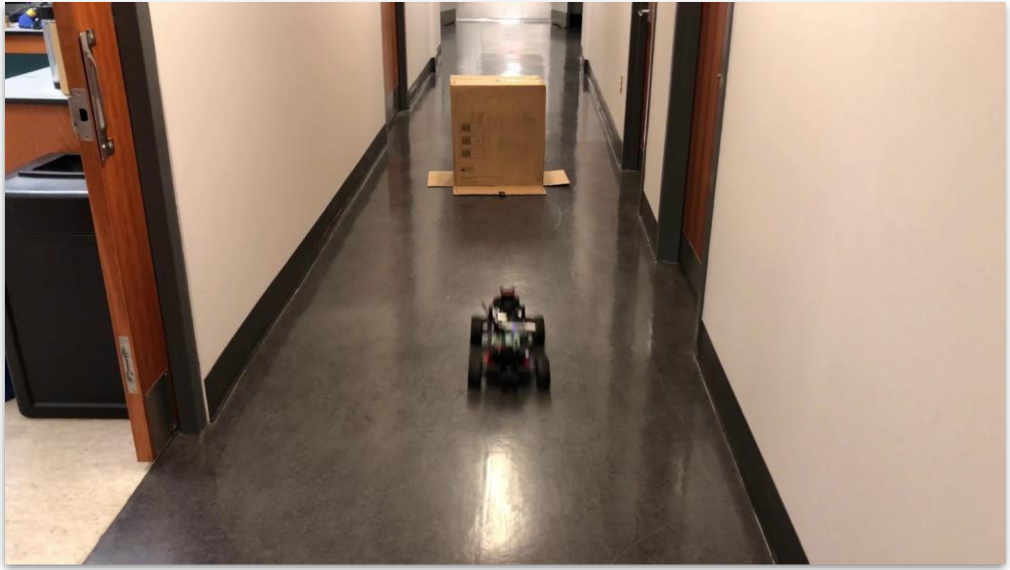
...we'll go through them...

Automatic Emergency Braking

Challenge:
Prevent the car from crashing while trying new algorithms.

Learning outcome:
Real-life implementations, sensors, failure modes.

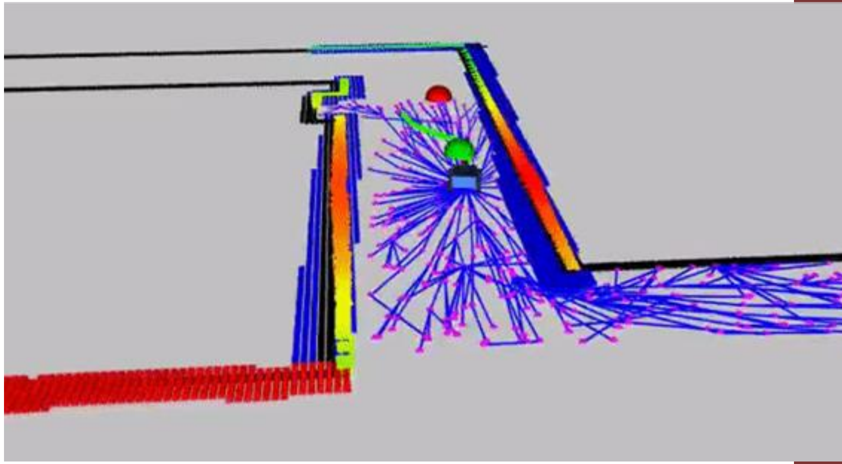
Assignment: Time-to-collision based braking



F1/10 simulator

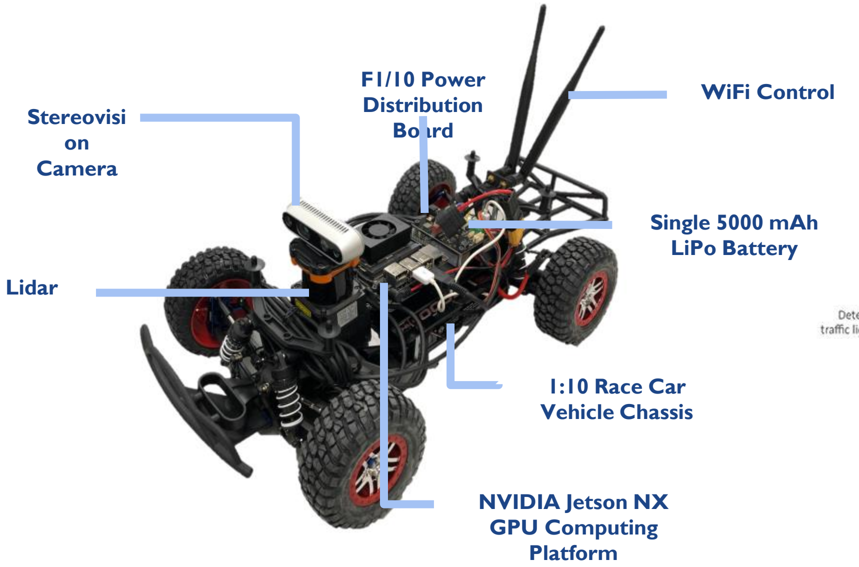
F1TENTH Gym:

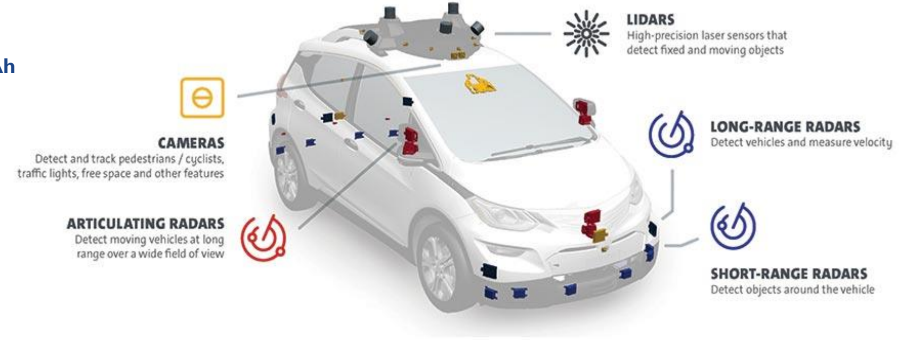
- Lightweight 2D simulator built in Python
- Asynchronous
- Faster than real-time execution (30x realtime)
- Realistic vehicle simulation and collision
- Runs multiple vehicle instances
- Publishes laser scan and odometry data
- Built for fast prototyping



Pose Representation and Transforms

Each sensor provides measurements in the frame of reference specific to that sensor





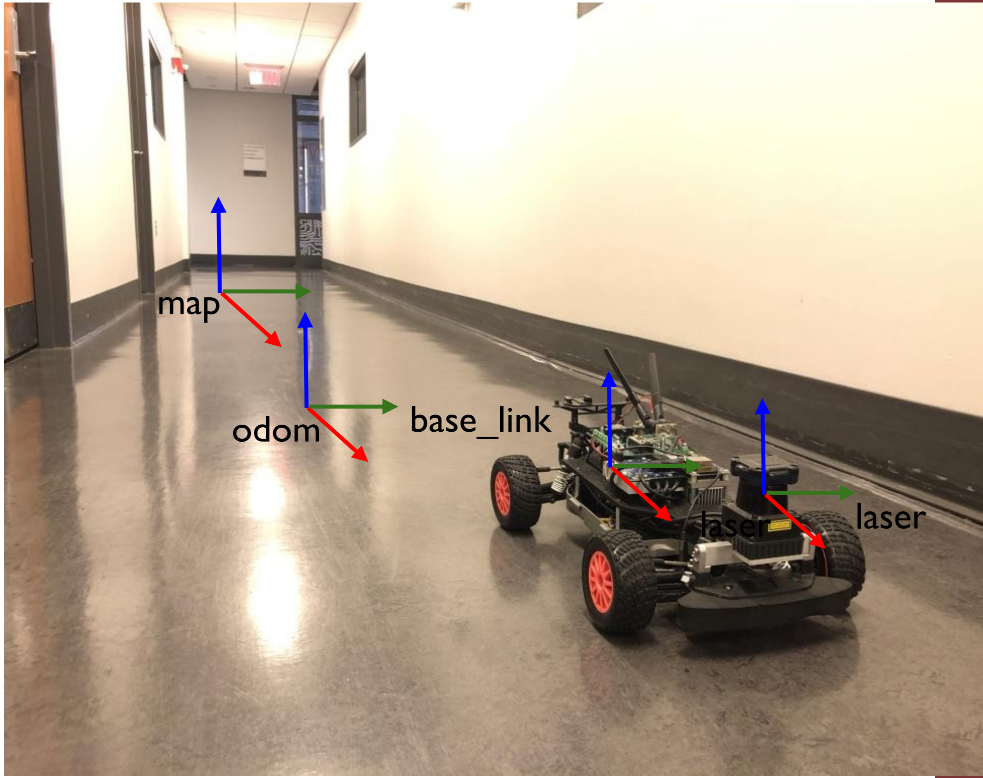
www.autonomousvehicletech.com

Pose Representation and Transforms

Challenge:
Sensors in different reference frames

Learning outcome:
Coordinate frames, Rigid body transforms

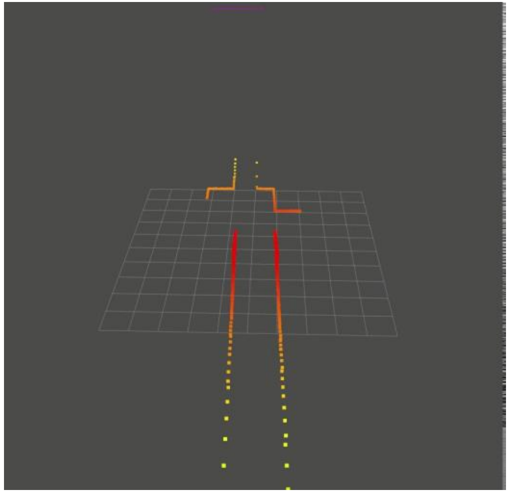
Assignment:
Pose transformations in ROS

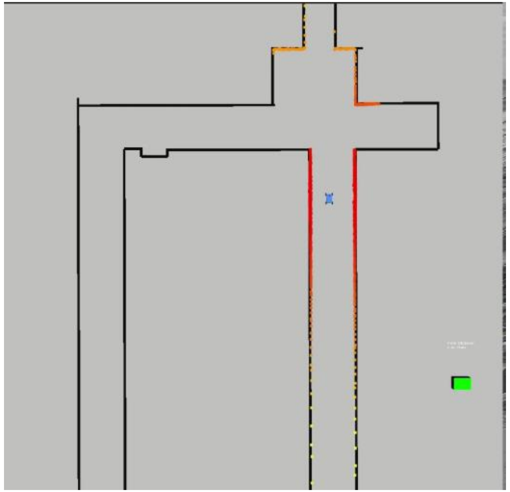


Multiple Reference Frames

The world makes more sense if we put the laser scans in the global map frame instead of the **laser frame.**

More information available when planning in a **global frame** instead of extracting information in the observation frame.





Wall-following

Challenge:
How can we drive the car around the track

Learning Outcome:
Basics of PID, how to compute error, failure modes.

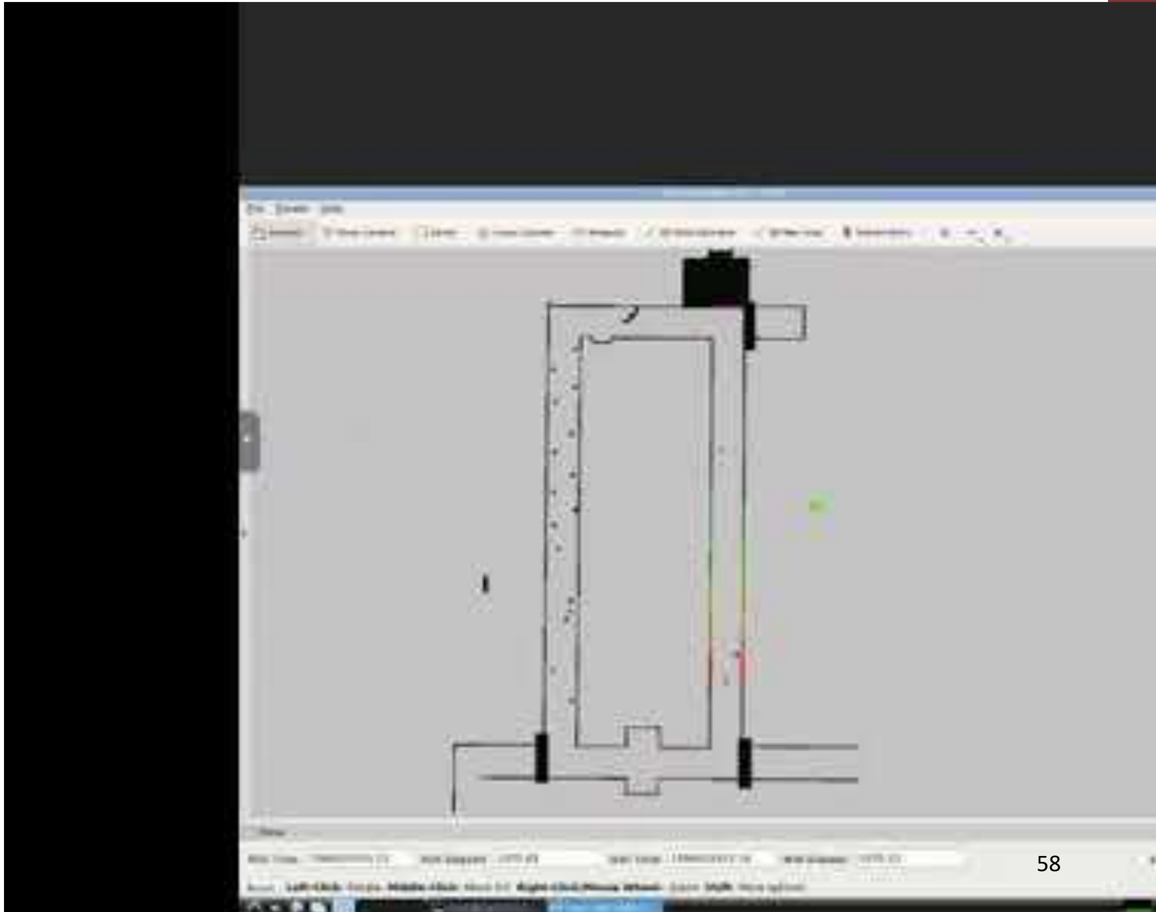
Assignment:
Wall following in simulation and on the vehicle.

Obstacle Avoidance: Follow the Gap

Challenge:
How can we avoid obstacles

Learning Outcome:
Basics of reactive navigation, avoidance on both static and dynamic obstacles

Assignment:
Follow the gap in simulation and on the vehicle.



Obstacle Avoidance: Follow the Gap

Challenge:
How can we avoid obstacles

Learning Outcome:
Basics of reactive navigation, avoidance on both static and dynamic obstacles

Assignment:
Follow the gap in simulation and on the vehicle.

Race 1: Reactive Methods

Race Format: Time attack, single car

Penalties: Crashing

Baseline: Complete 5 laps without crashing

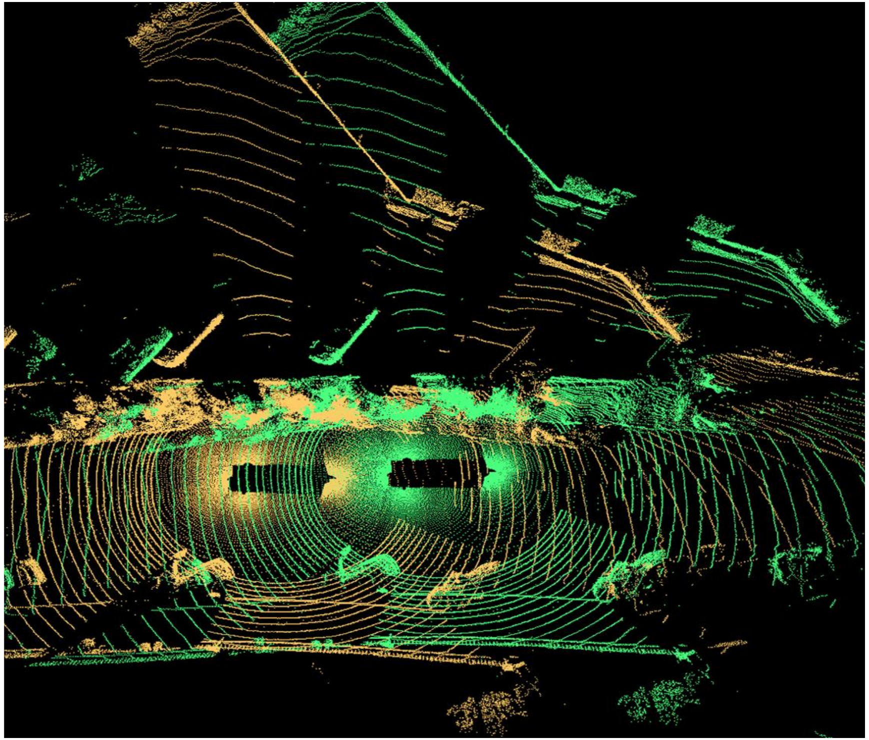
Example Video: Follow the Gap in CPSWeek Grand Prix'18

Localization: Scan Matching

Challenge:
Where is the robot with respect to the previous frame

Learning Outcome:
Iterative closest point algorithm, implementing a real research paper

Assignment:
Scan matching using iterative closest point in the simulator

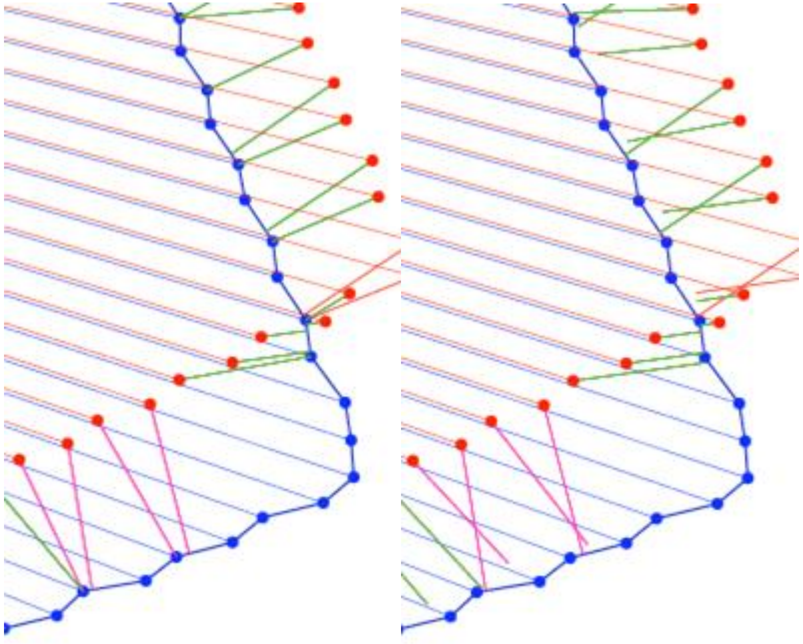


Localization: Scan Matching

Scan matching is a fundamental localization algorithm, and is used in most of the modern SLAM algorithms.

Students implement different metrics (point-to-point vs. point-to-line, shown on the right) for ICP from a research paper.

Students implement fast correspondence search to make ICP practical for a moving robot.

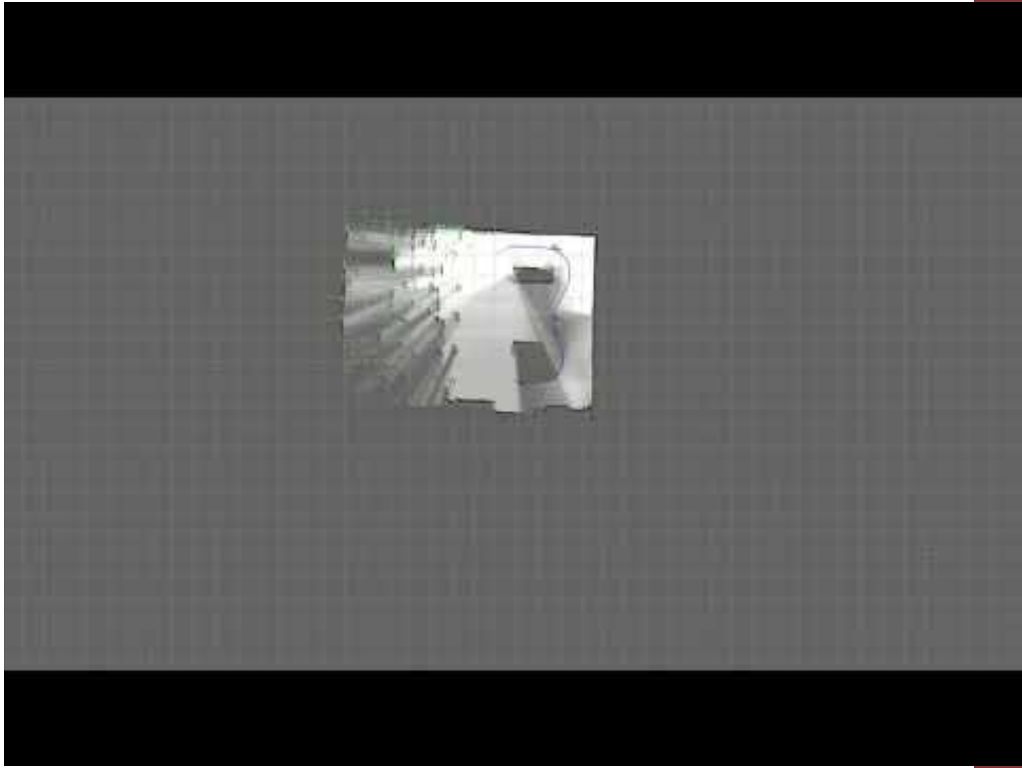


Simultaneous Localization and Mapping with Cartographer

Challenge:
How to use state-of-the-art tools for map building.

Learning Outcome:
Understanding the Cartographer paper and how it relates to scan matching.

Assignment:
Build maps with Cartographer of race track on the car.

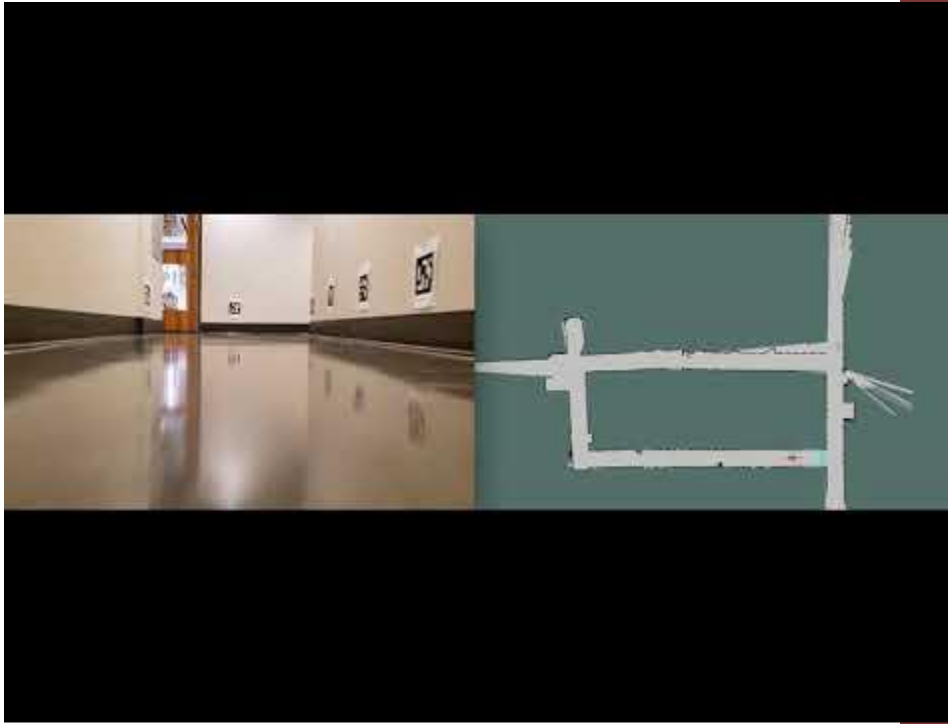


Localization: Particle Filter

Challenge:
Given a map of the world and multiple sensor observations, what is the pose of my robot?

Learning Outcome:
Understanding particle filter, which is a version of a bayesian filter

Assignment:
Running Particle Filter to localize in the world

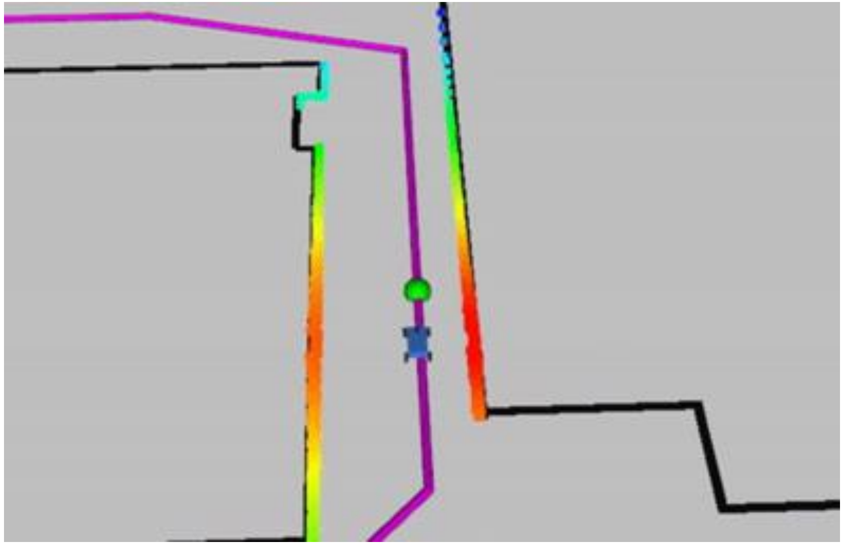


Pure Pursuit

Challenge:
How to track a reference trajectory given a map the ability to localize?

Learning Outcome:
Closed form geometric approach and alternatives.

Assignment:
Implement pure pursuit waypoint tracker in the simulator and on the car.



Pure Pursuit

Challenge:
How to track a reference trajectory given a map the ability to localize?

Learning Outcome:
Closed form geometric approach and alternatives.

Assignment:
Implement pure pursuit waypoint tracker in the simulator and on the car.

Race 2: Map-based Methods

Race Format:
Time attack, single car

Penalties:
Crashing

Baseline:
Complete 5 laps without crashing

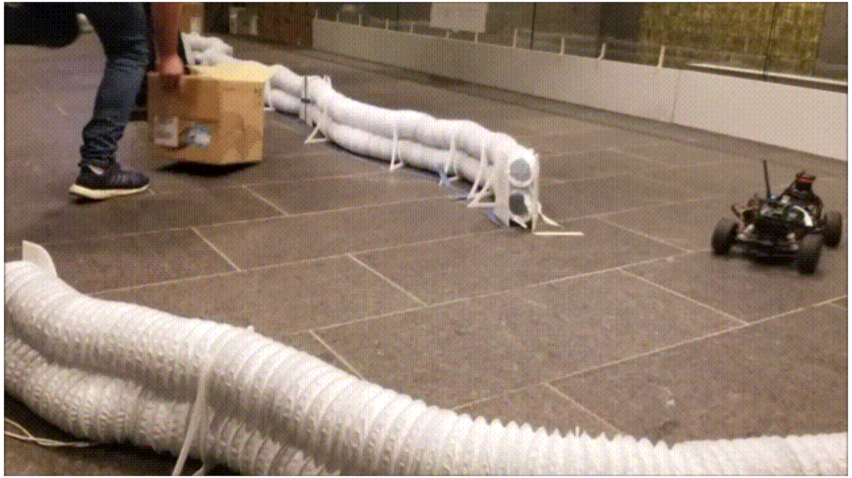
Example Video:
Pure Pursuit CPSWeek 2019

Motion Planning

Challenge:
How do we combine the capabilities of map based methods while being able to avoid obstacles

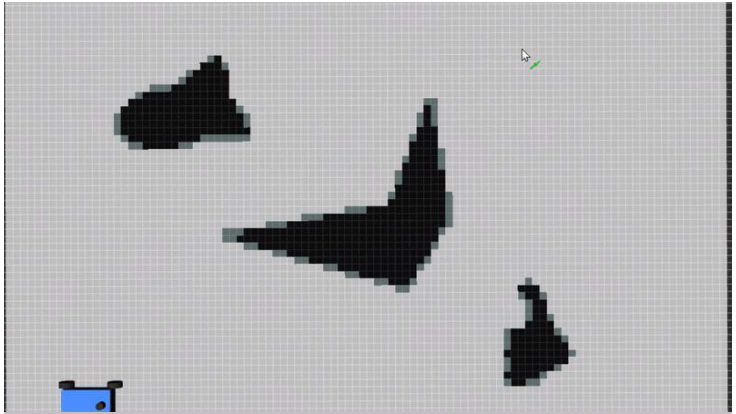
Learning Outcome:
Understanding search-based motion planning, probabilistic planning methods, RRT and its variants

Assignment:
Implement RRT in the simulator and on the car.



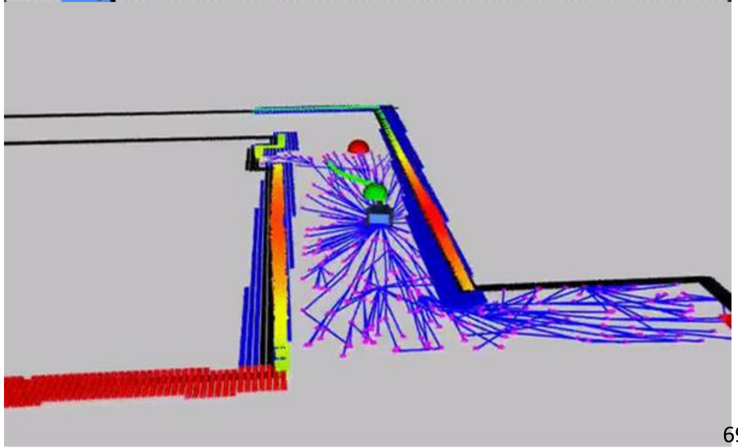
Motion Planning

Occupancy grid: approximating the real world with a discrete representation, also relates back to the SLAM lecture



Planning in discrete space with search-based planning methods (A*, Dijkstra's)

Planning in continuous space with probabilistic planning methods (RRT, RRT*)

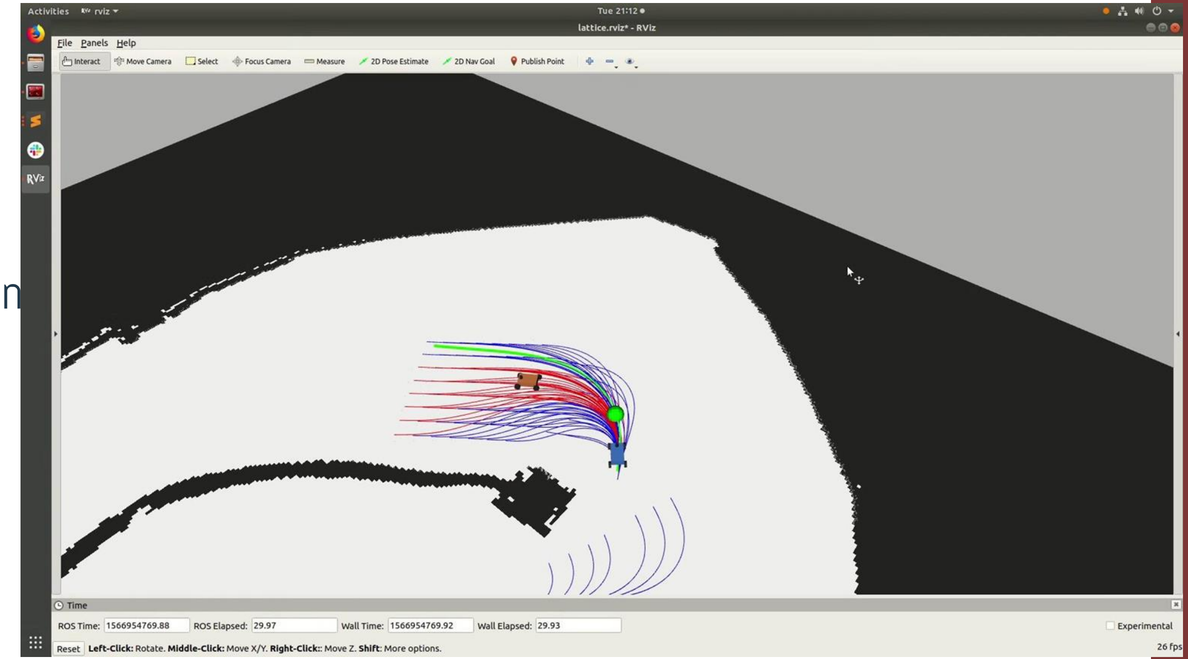


Model Predictive Control

Challenge:
Create dynamically feasible trajectories for overtaking

Learning Outcome:
Trajectory optimization & samplin
based MPC

Assignment:
As a project option



Module E: Learning and Vision

Detection and Pose Estimation: classical methods

Challenge:

Where is the other car?

Learning Outcome:

Understanding camera model, single view geometry, Homography, detecting features, and prediction.

Assignment:

Camera calibration, detecting poses of AprilTags, predicting the trajectory of adversarial vehicle.

Detection and Pose Estimation: ML-based

Challenge:

Where is the other car without using fiducial markers?

Learning outcomes:

Understanding multi-view geometry, the epipolar constraint, stereo vision, and using Convolution Neural Network detectors.

Assignment:

Making the detection pipeline fast

Reinforcement Learning

Challenge:

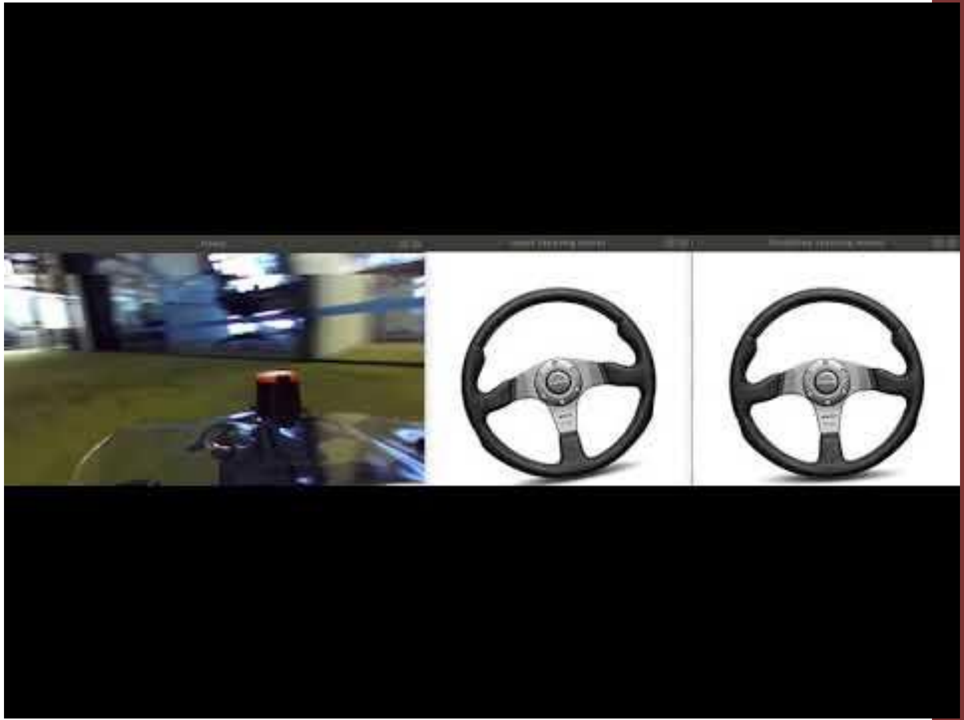
How to learn from human drivers?

Learning outcomes:

Understand imitation learning and implement it

Assignment:

RL as a project option.



Race 3: Head-to-Head

Race Format: Round-robin, two cars

Penalties: Crashing

Baseline: Avoid static obstacles

Example Video: Follow-the-gap
CPSWeek'19

BUILD CODE RACE F1TENTH CREW GALLERY FORUM

f1tenth.org