F1tenth and Autonomous Driving an Introduction

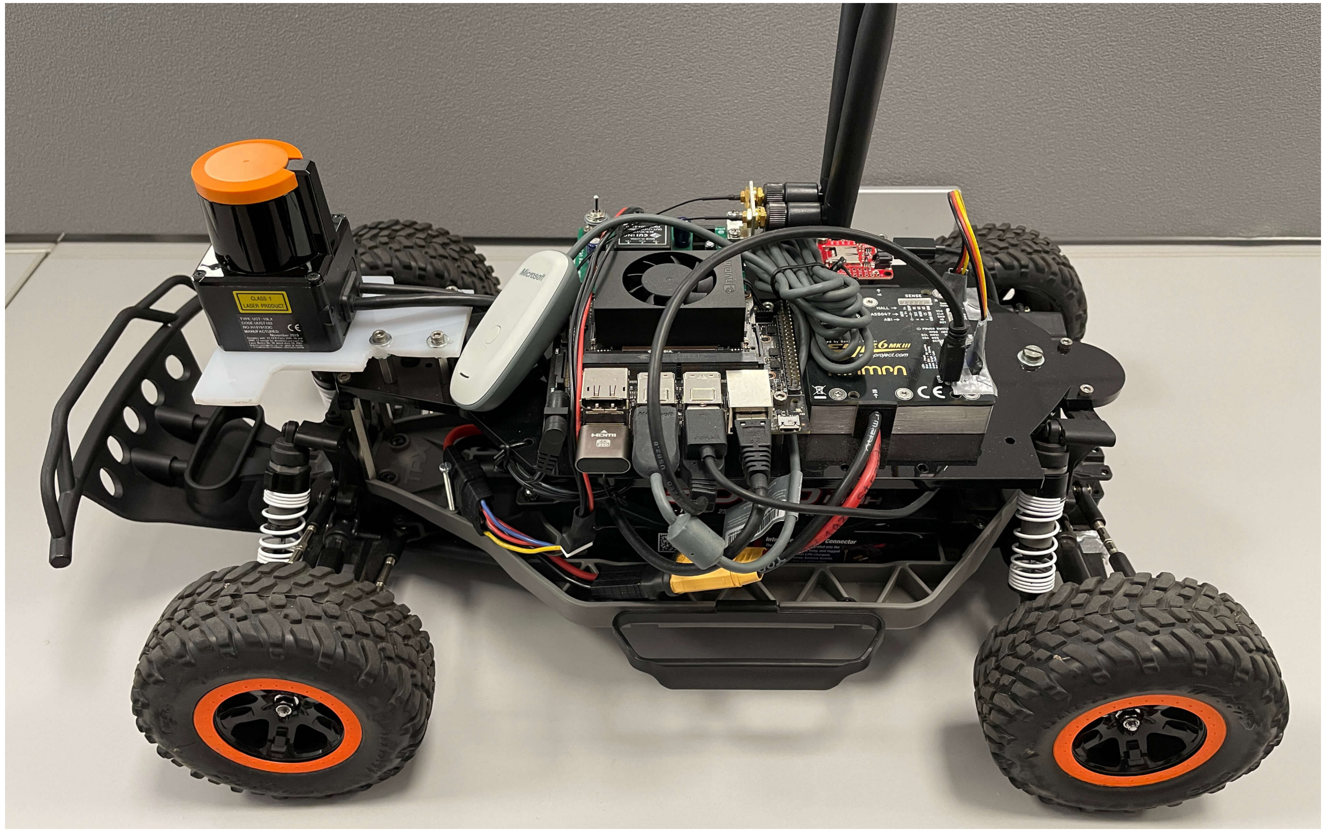
Michael Bosello

michael.bosello@unibo.it

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Smart Vehicular Systems A.Y. 2021/2022

F1TENTH: 1/10th scale Autonomous Racing

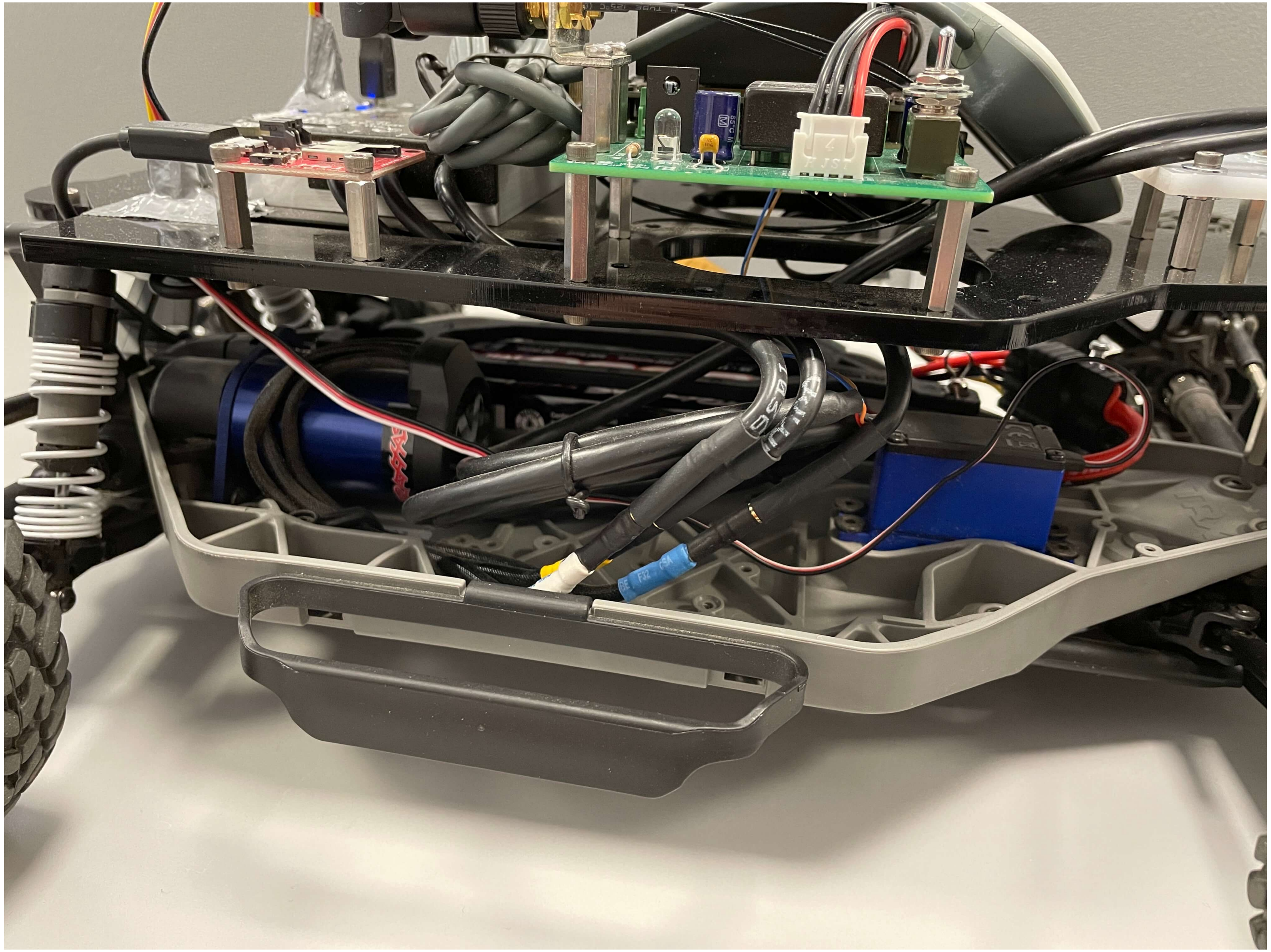


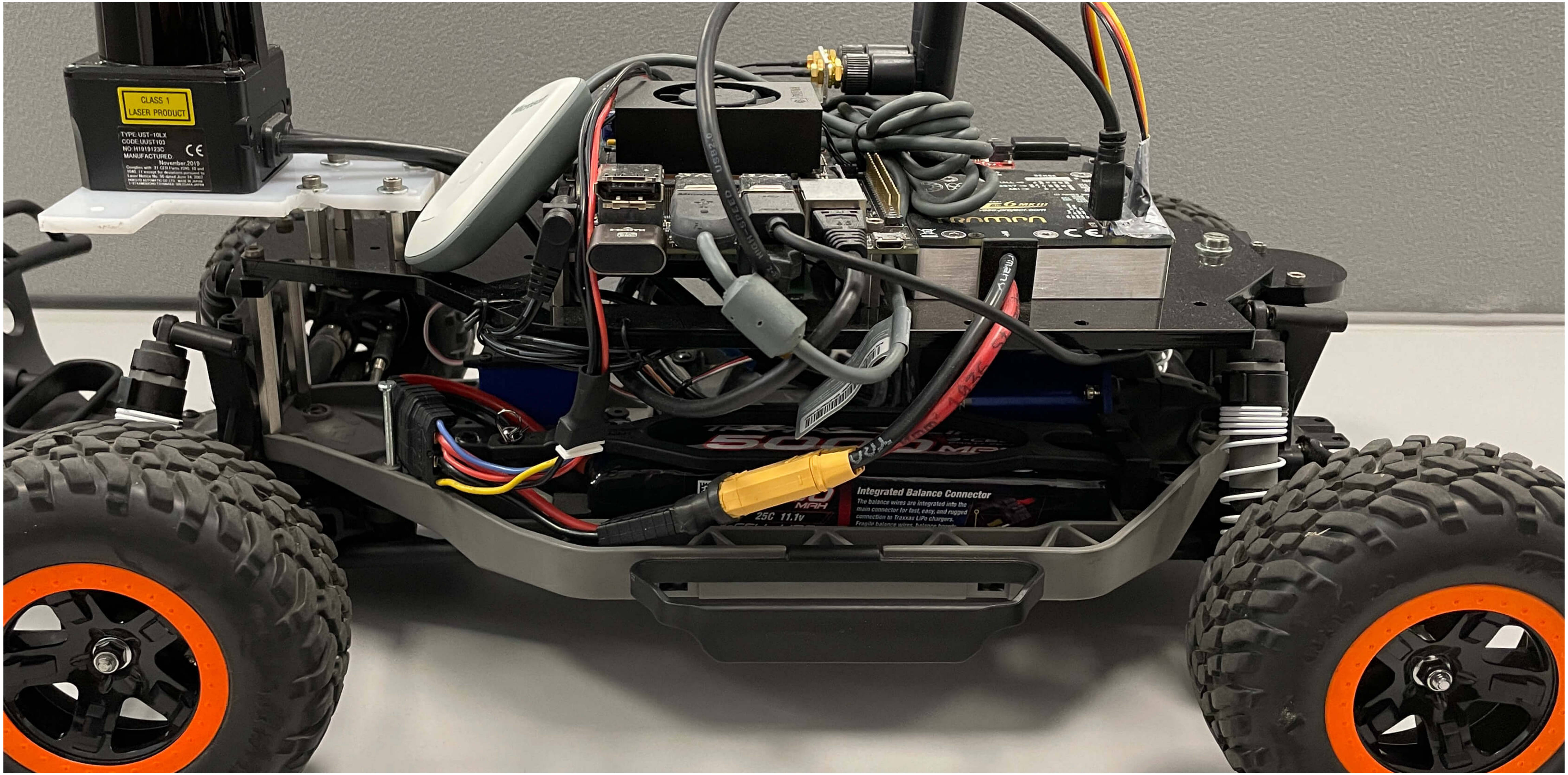
- The agent still faces challenges of a real driving scene
- Inexpensive
- Safe

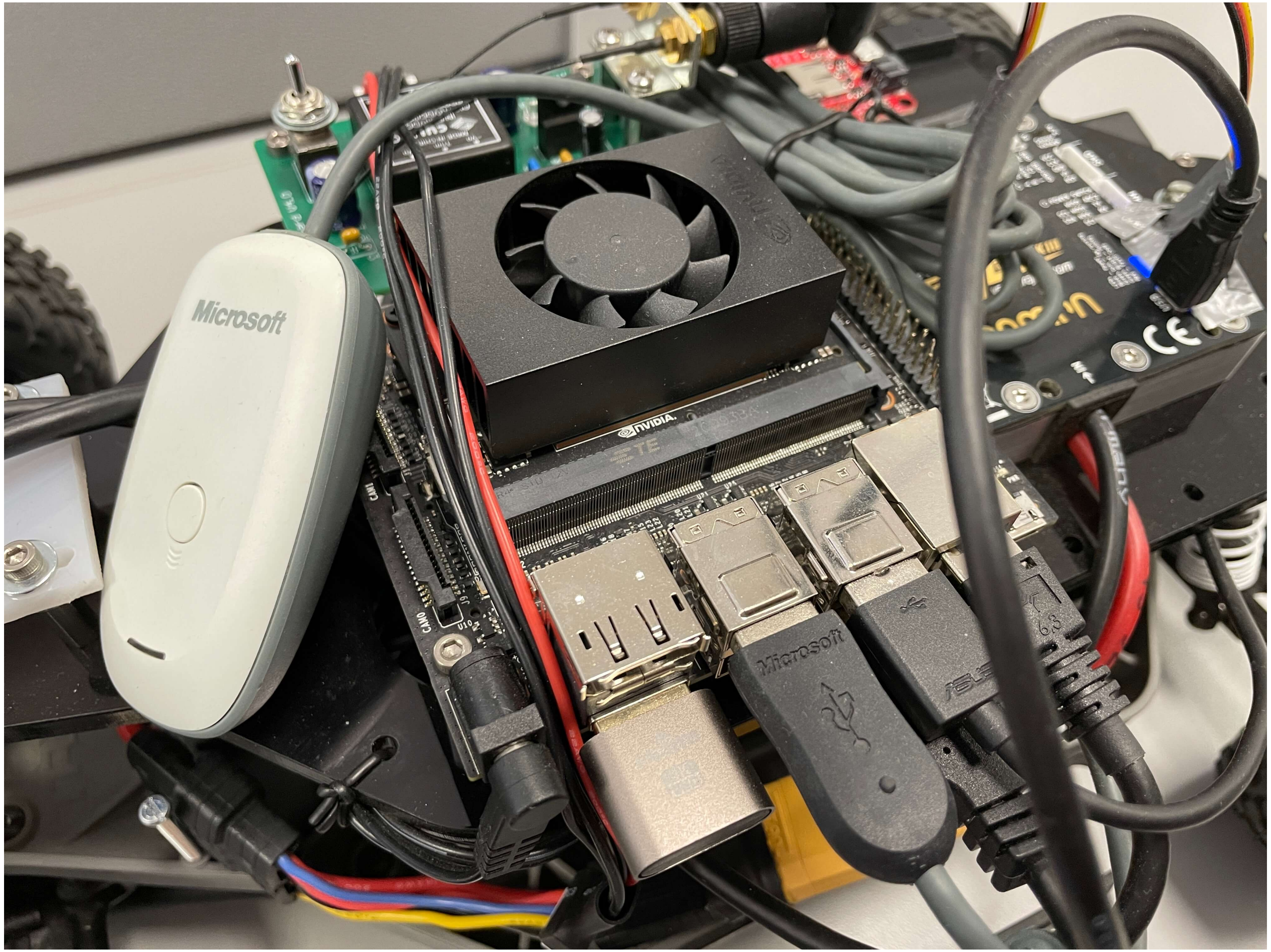
Very realistic 1/10 scale car prototype

- Hardware/software stacks similar to full-scale solutions
- Ackermann steering
- High speeds

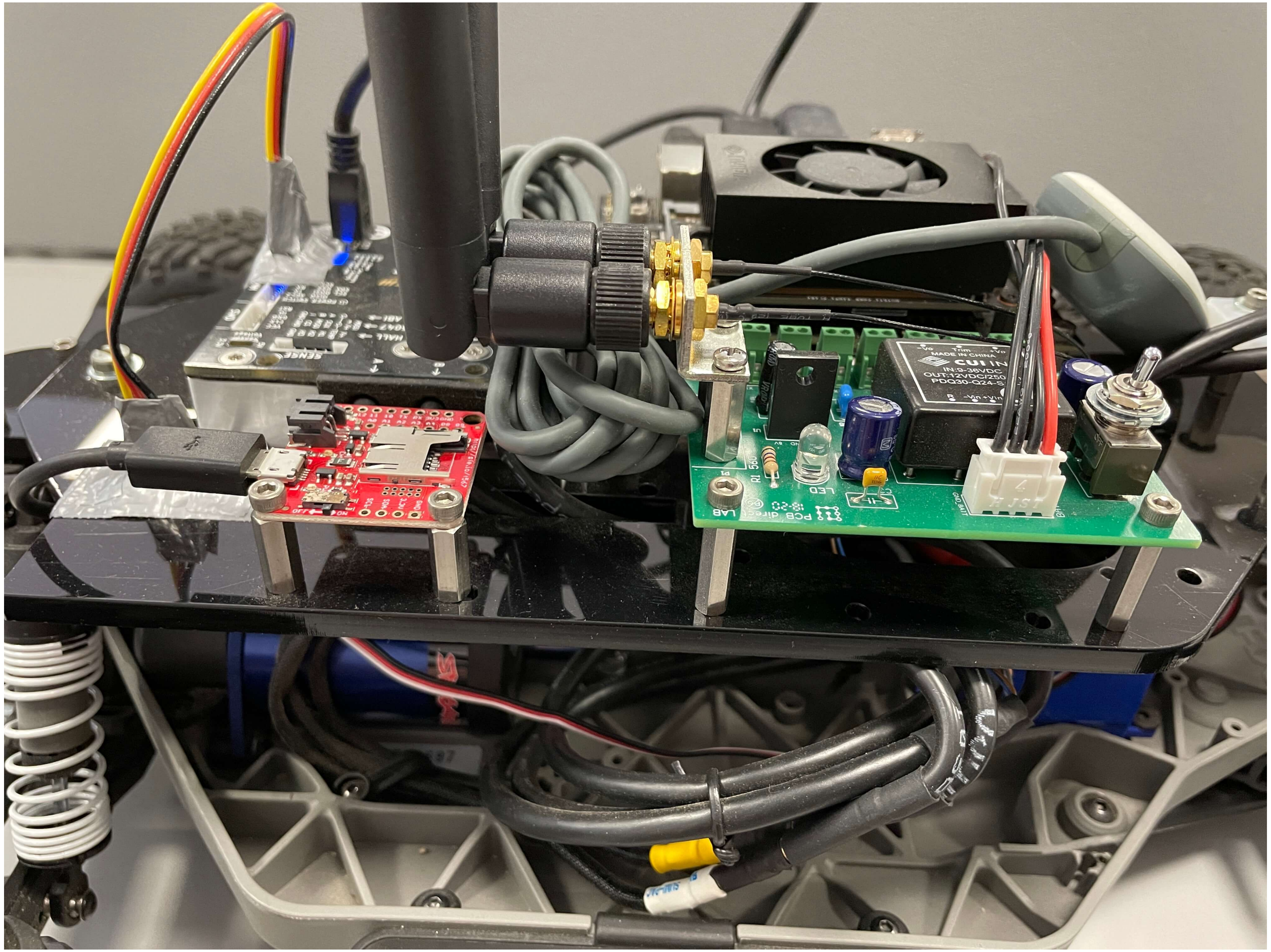
The same code can run on the ad-hoc simulator




CLASS 1
LASER PRODUCT
TYPE:UST-10LX
CODE:UUST103
NO:H1919123C
MANUFACTURED:
November 2019
Integrated Balance Connector
25C 11.1v

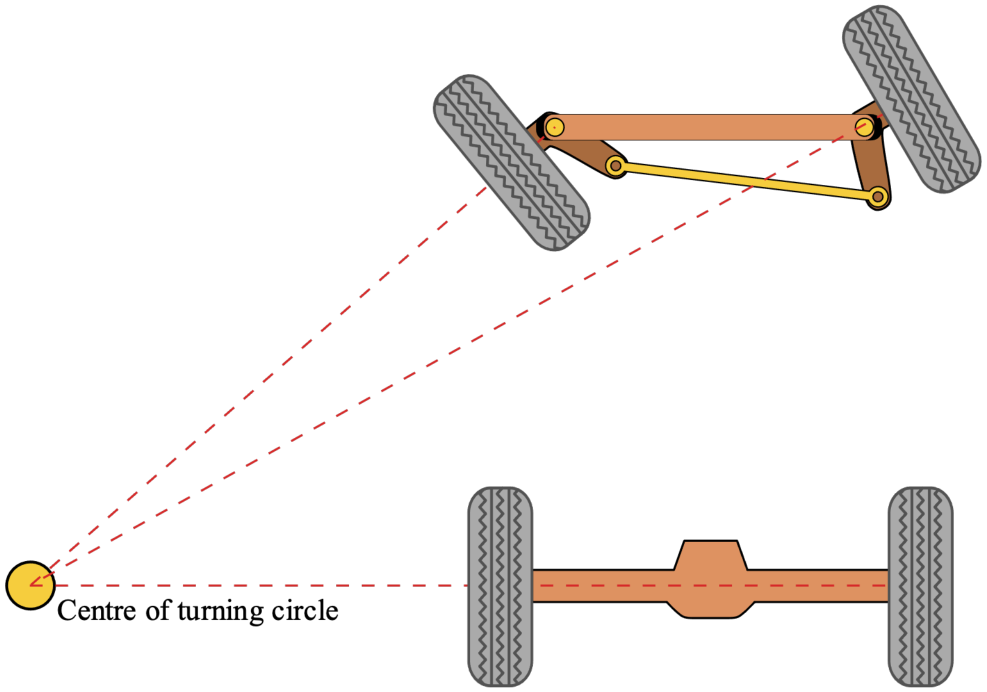

Microsoft
NVIDIA
Microsoft

Designed by Benjamin Vedder
VESC 6 MK III
vesc-project.com
TRAMPA
SENSE
HALL
Sensor Voltage
5V
3.3V
VCC
CLK
GND
IO
RST
SWD
POWER SWITCH
ADC2
TX
RX

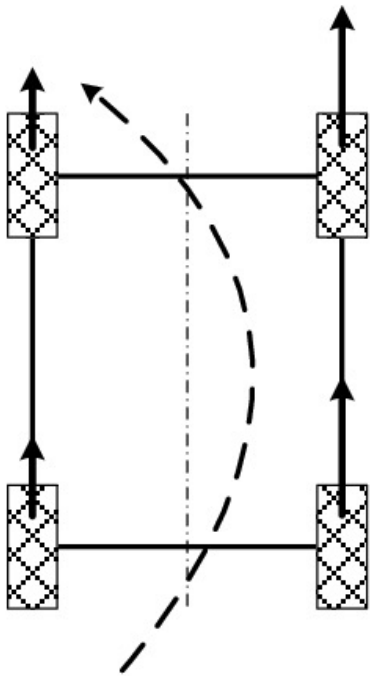


MADE IN CHINA
IN:9-36VDC
OUT:12VDC/250
PDQ30-Q24-S
LED

Steering

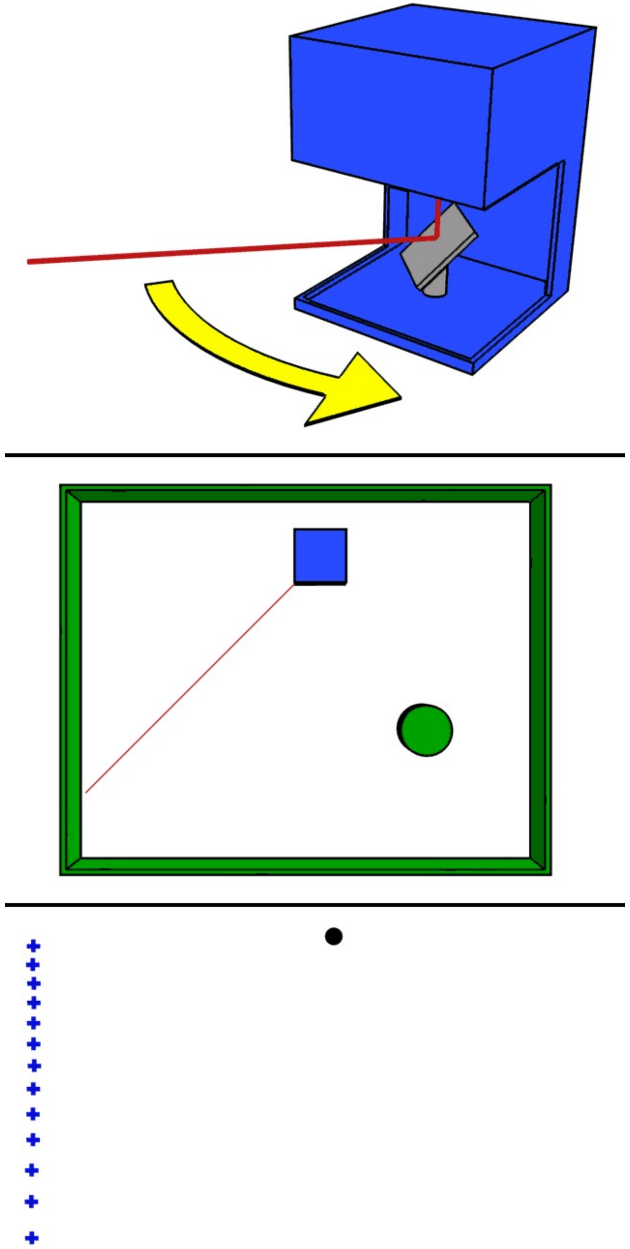


Ackermann steering
- 1 motor
- 1 servo



Differential steering
- 2 motors

2D LIDAR (Light Detection And Ranging)

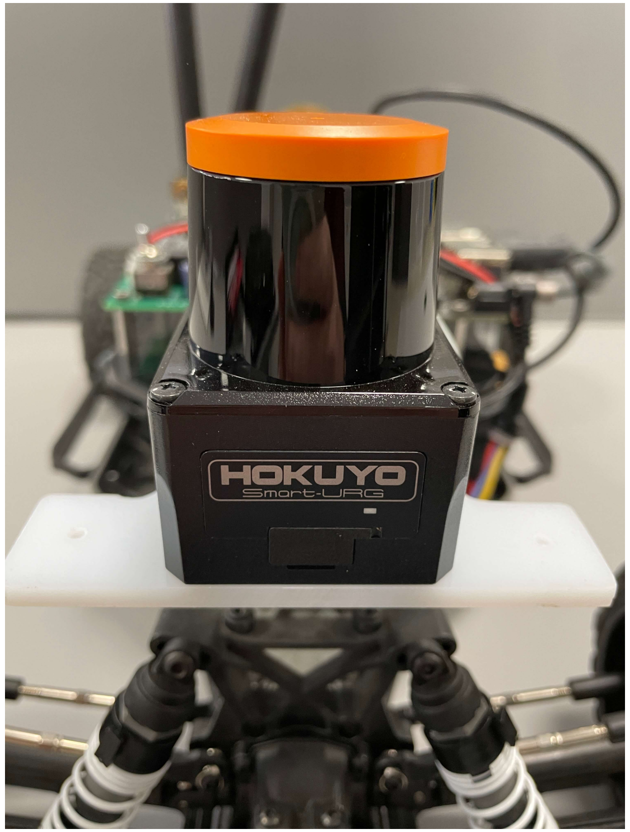


Source: Wikipedia

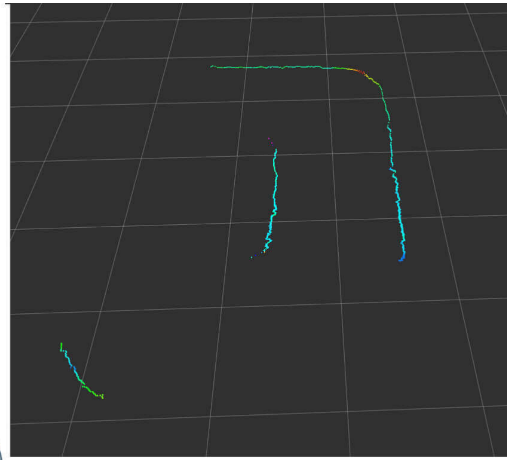
Sensor for measuring distances

- Emits focused light beams
- Measure the time of flight

- 2D: map of the azimuth at a fixed height
- Produces a vector of distances/intensities
- High frequency

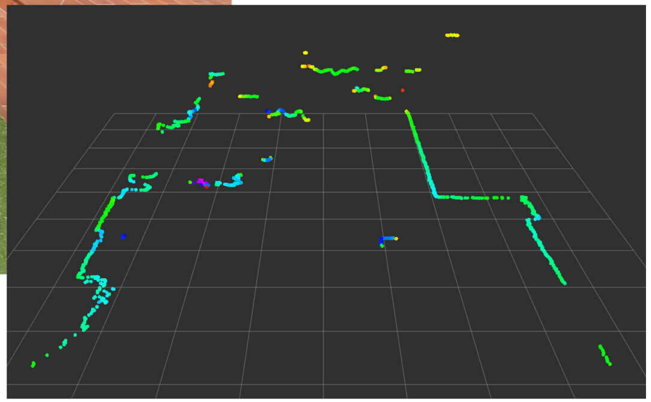


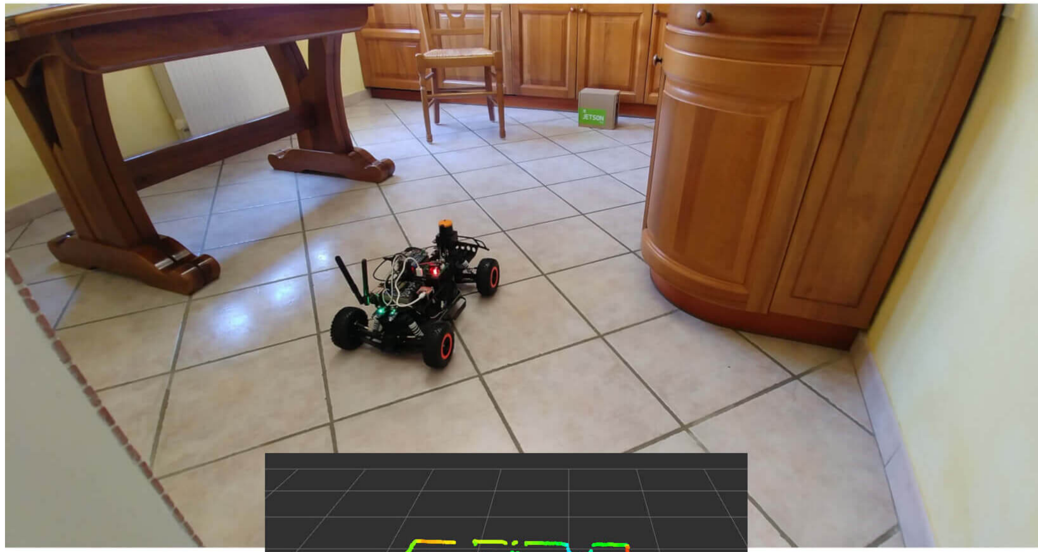
- LIDAR measurements are greatly affected by reflection
  - When a ray gets reflected, it appears as it is no obstacle in that direction

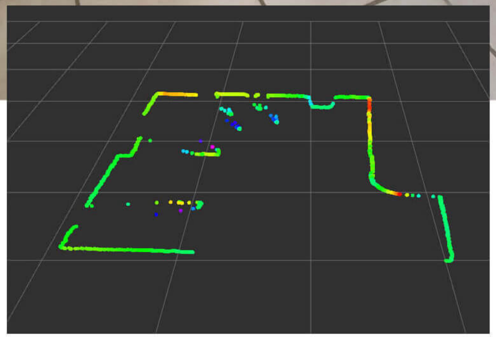


Hokuyo UST-10LX

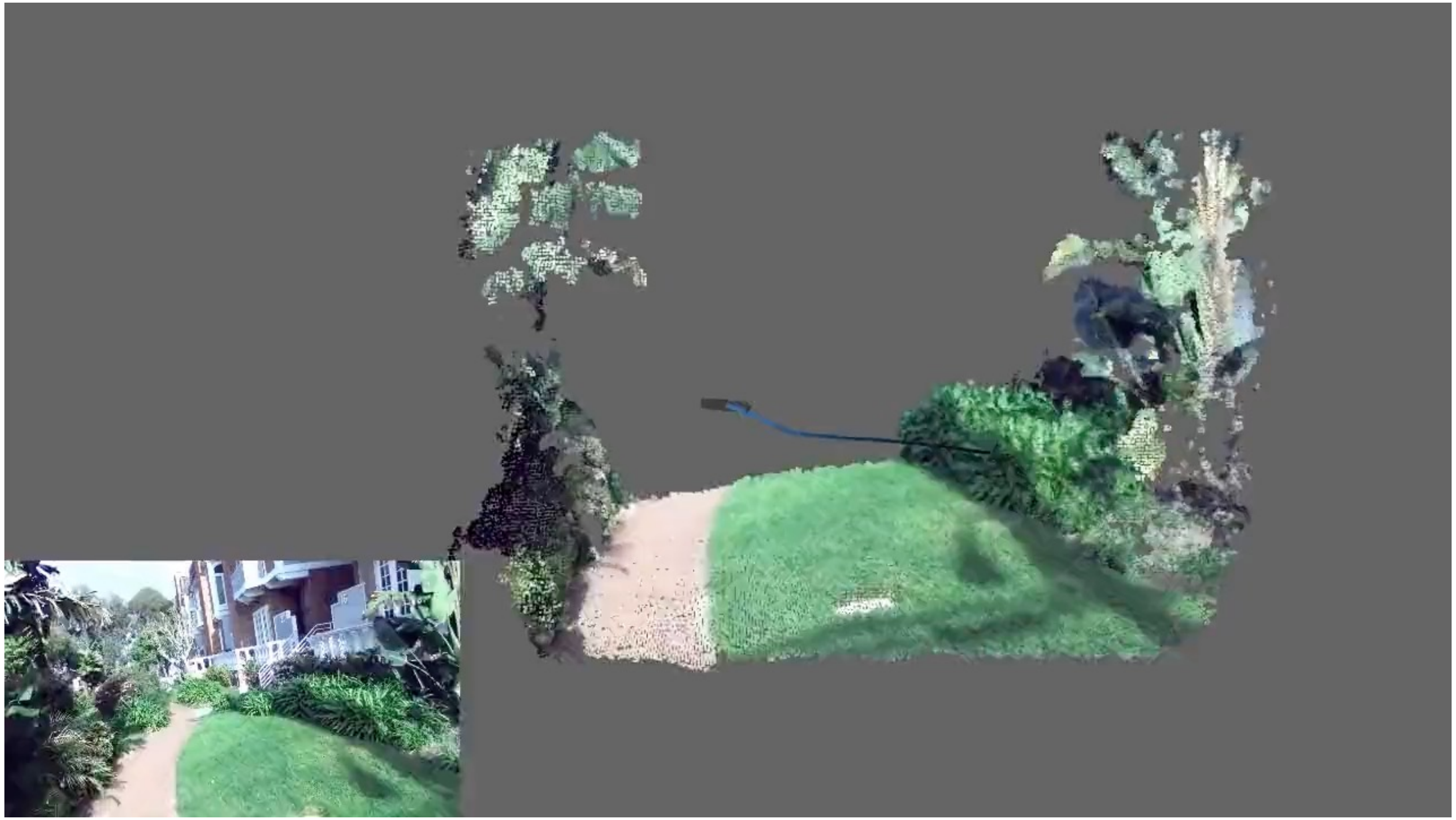
- -270° field of view
- -0.25° angular resolution
- -1081 scan rays
- -10m detection range
- -±40mm accuracy
- -25ms scan speed



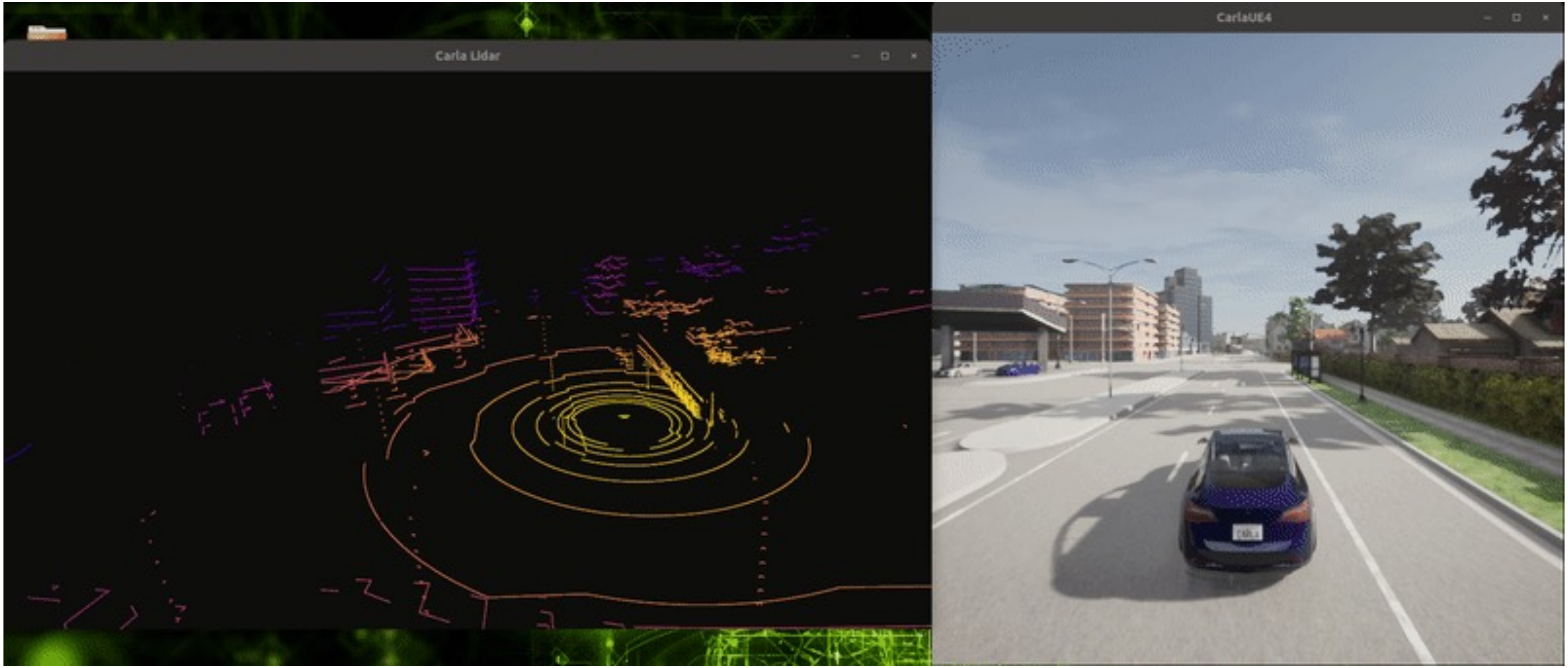




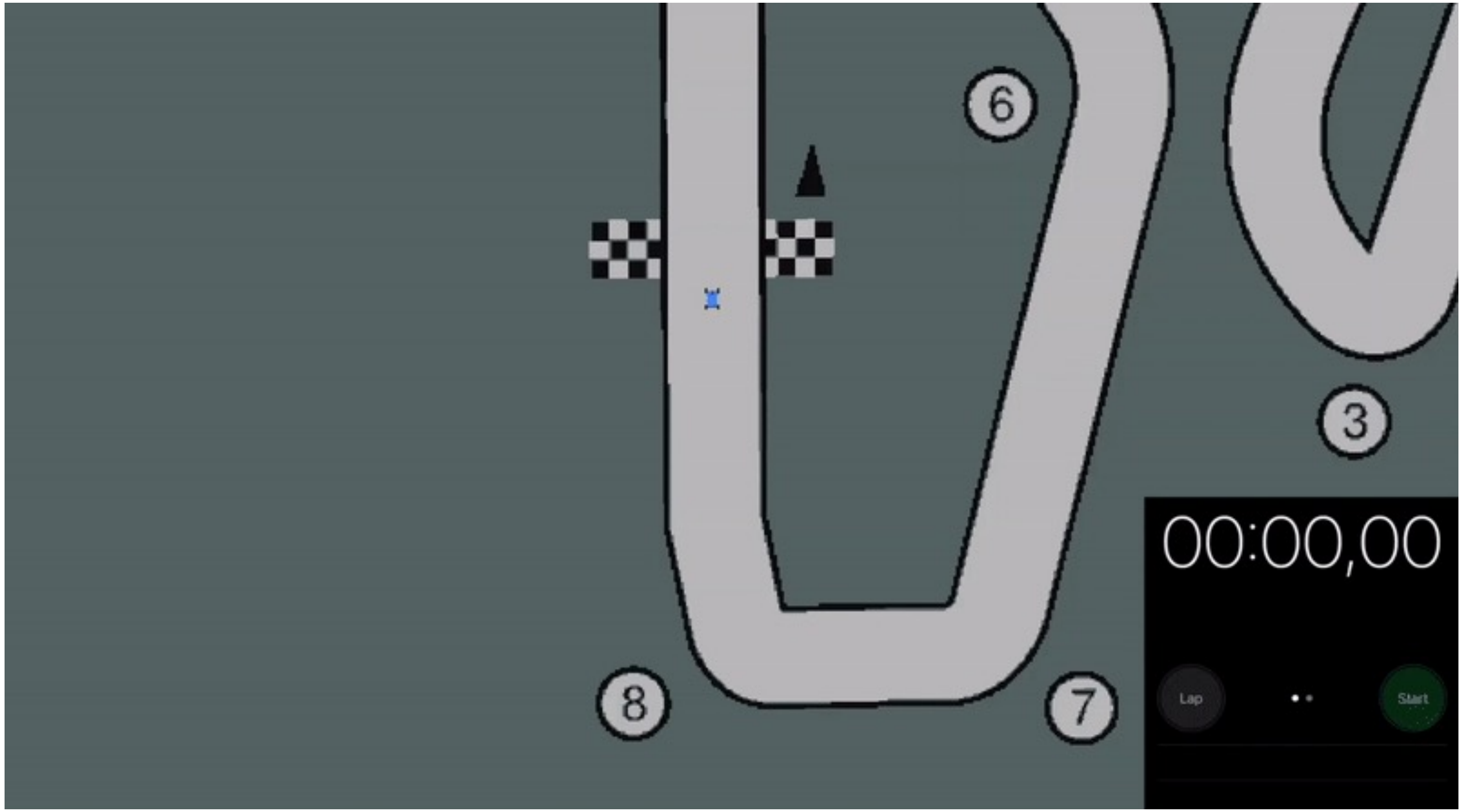
Stereo Camera



3D LIDAR



Simulator



- Simplified dynamics
- No sensors/actuators noises

Sim2real Problem

- Divergence from real dynamics
- Sensors/actuators noises
- High speed + embedded system = resource constrains

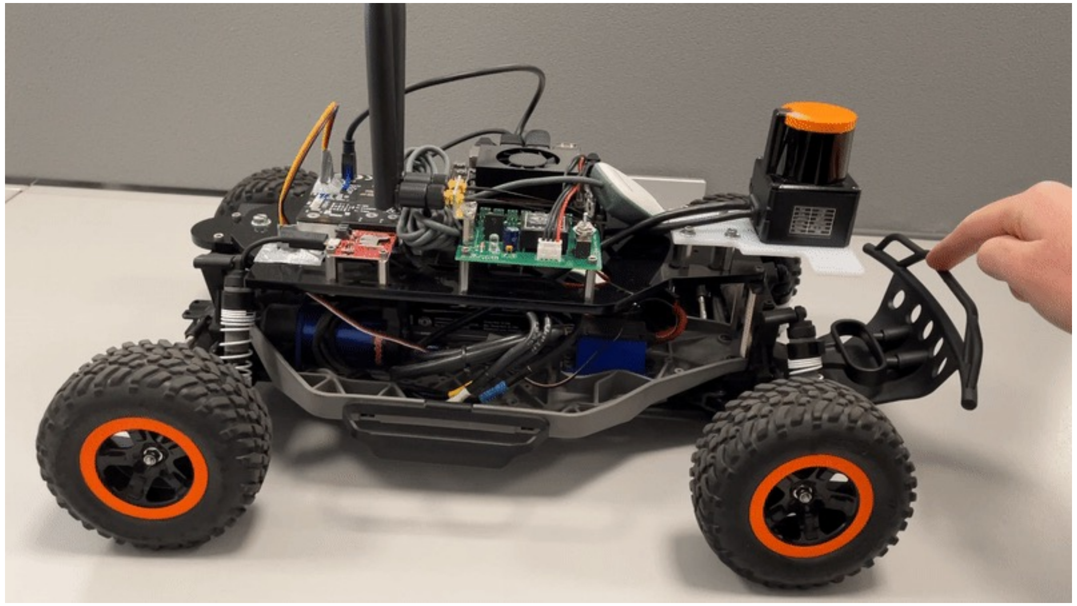
Classical control:

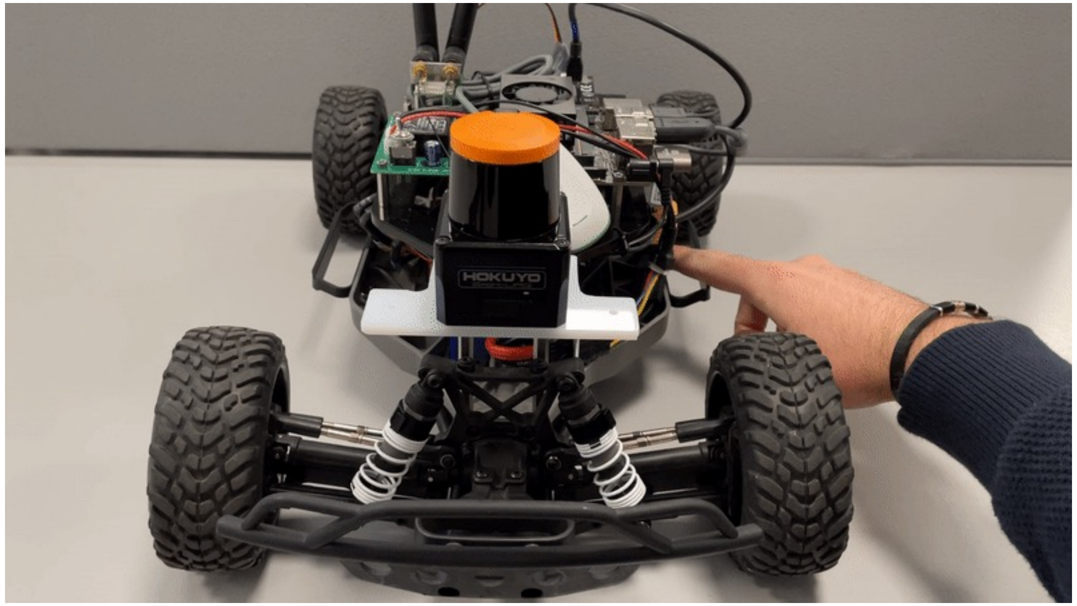
Simplified dynamics could lead to fail

Learning:

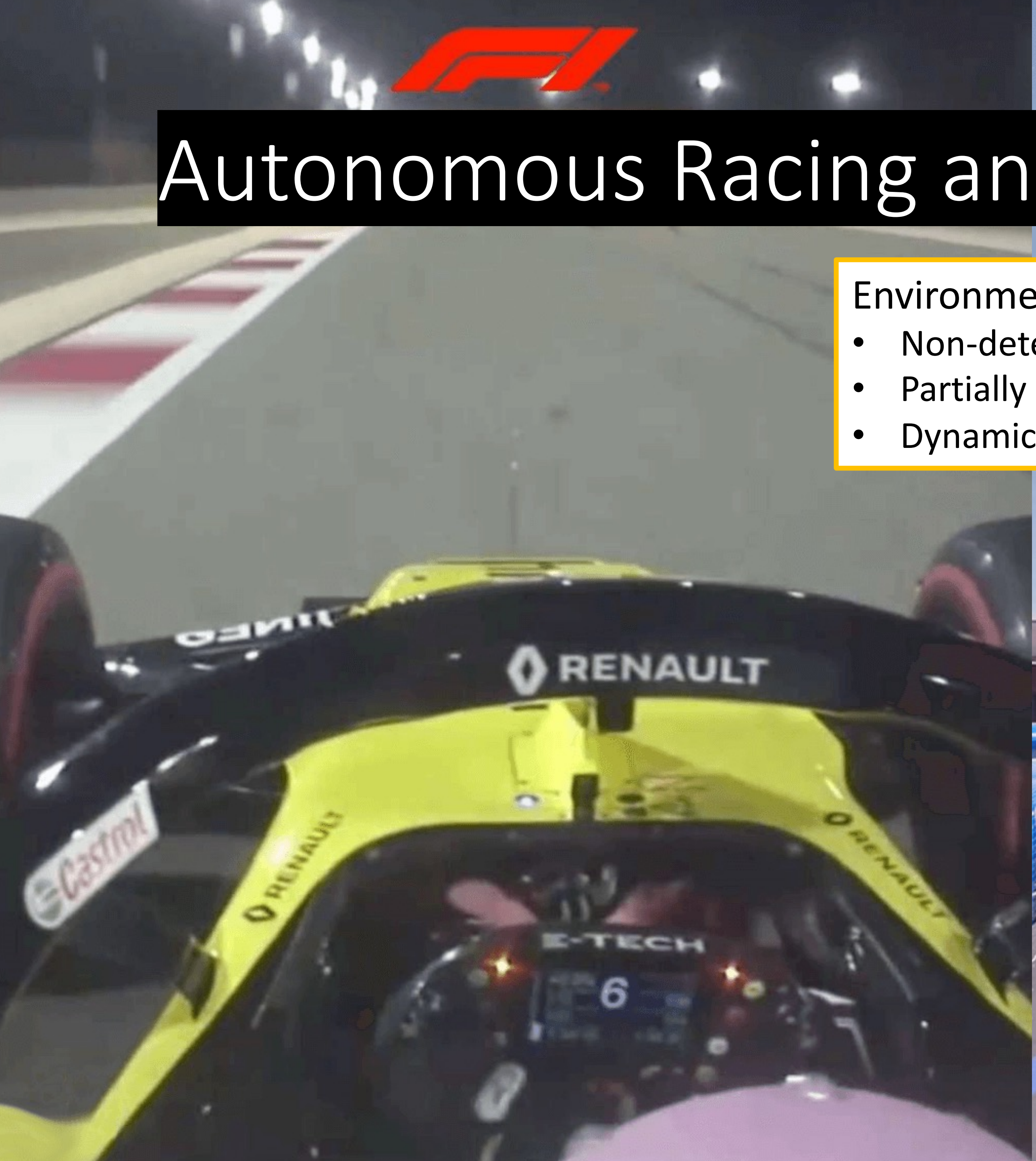
Data differs, e.g. image texture

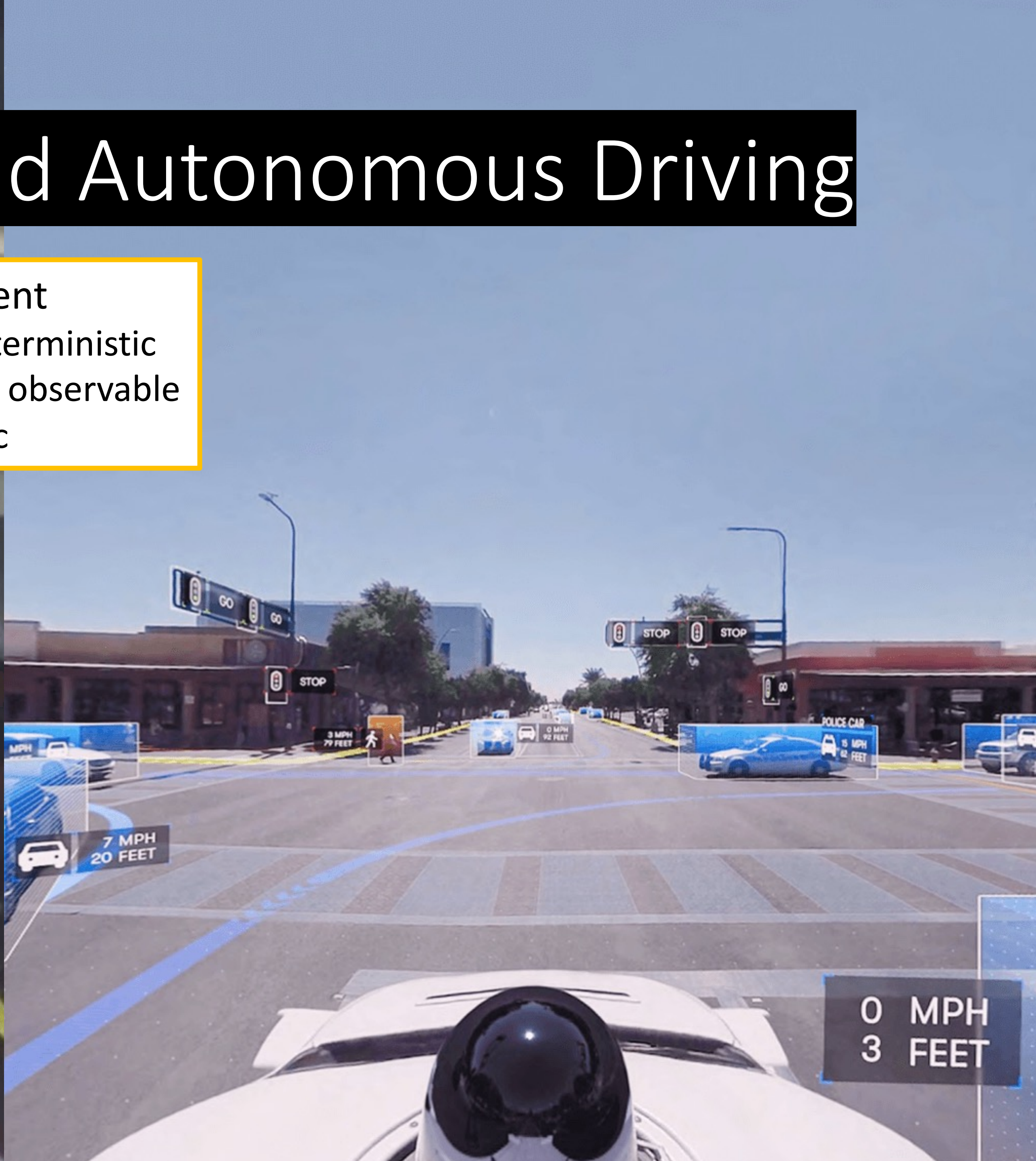
You need some form of transfer learning





Autonomous Racing and Autonomous Driving





Environment
- Non-deterministic
- Partially observable
- Dynamic

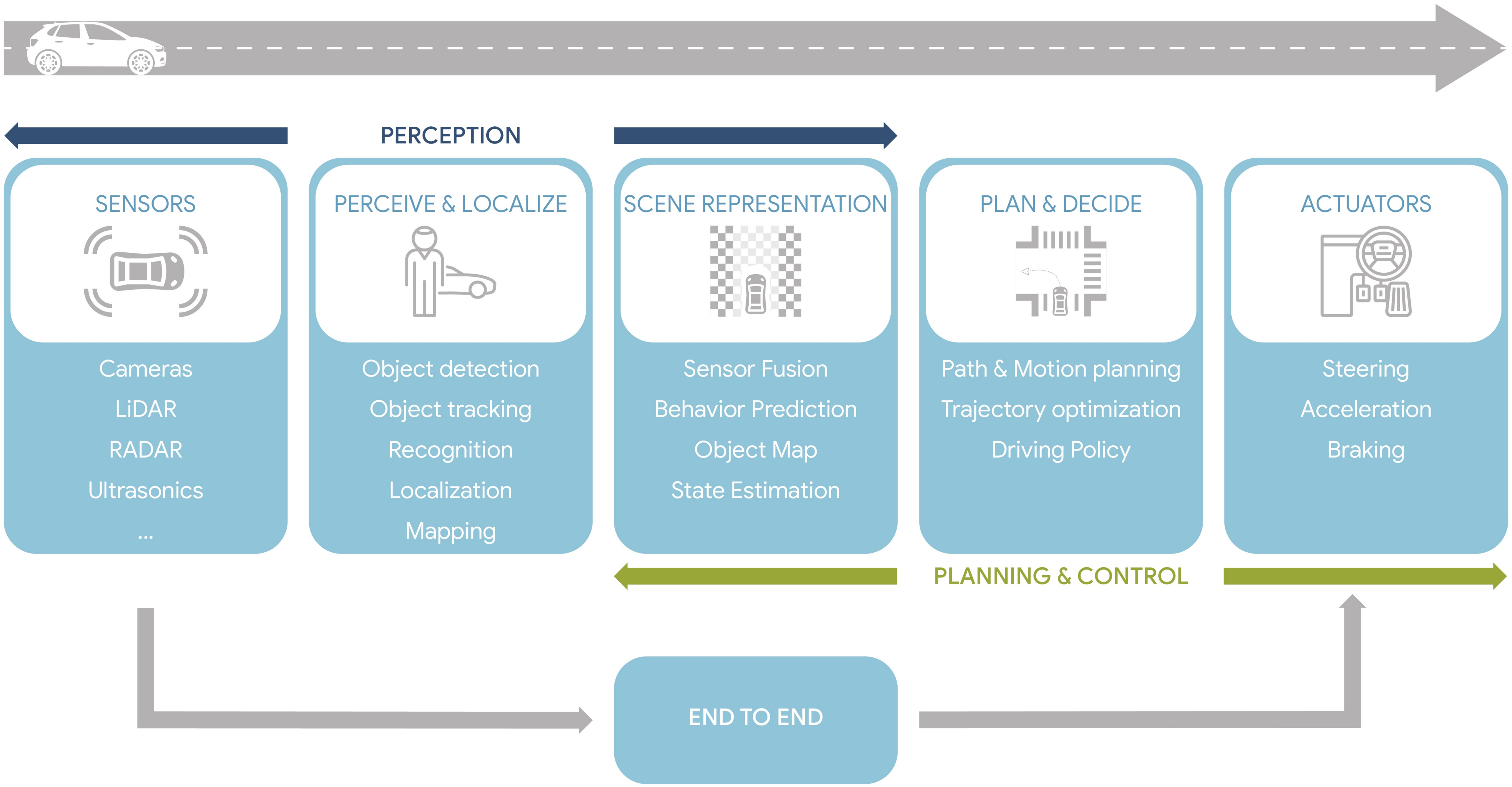
Indy Autonomous Challenge

The Race Problems

Perception. Planning. Control.

Approaches

- Single Task Handling
  - Use human ingenuity to inject knowledge about the domain
- End-to-end Learning
  - Let the algorithm optimize towards the final goal without constrains

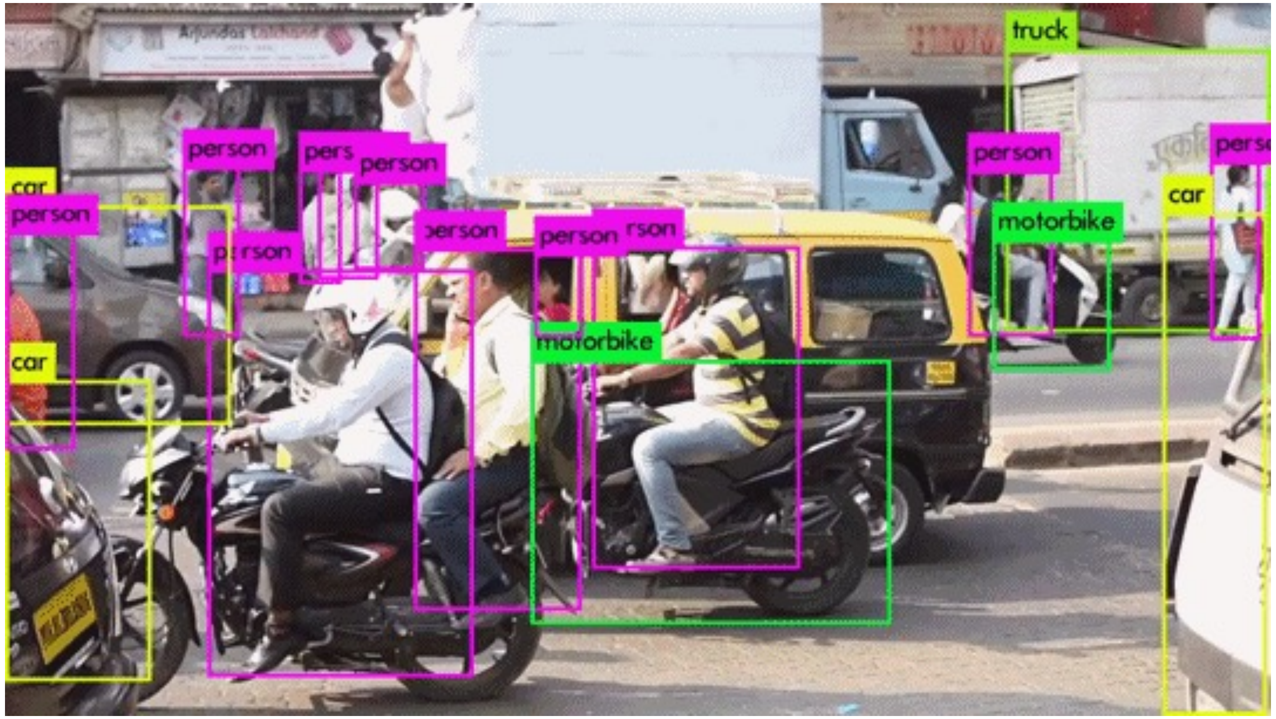

PERCEPTION
SENSORS
Cameras
LiDAR
RADAR
Ultrasonics
...
PERCEIVE & LOCALIZE
Object detection
Object tracking
Recognition
Localization
Mapping
SCENE REPRESENTATION
Sensor Fusion
Behavior Prediction
Object Map
State Estimation
PLAN & DECIDE
Path & Motion planning
Trajectory optimization
Driving Policy
ACTUATORS
Steering
Acceleration
Braking
PLANNING & CONTROL
END TO END

Perception: Recognition

Mainly based on Deep Learning

- Classical Computer Vision algorithms
  - Not robust enough
  - Slow

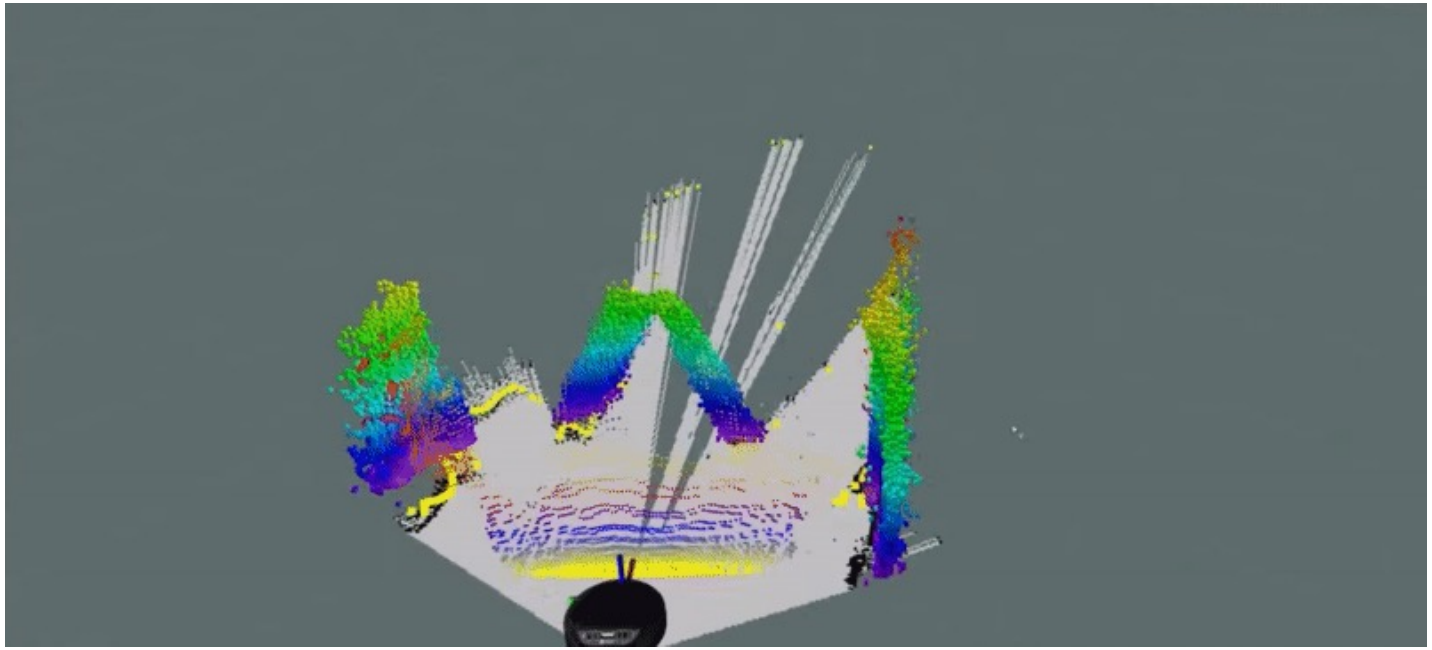
- Convolutional Neural Networks (CNNs)



YOLO v3

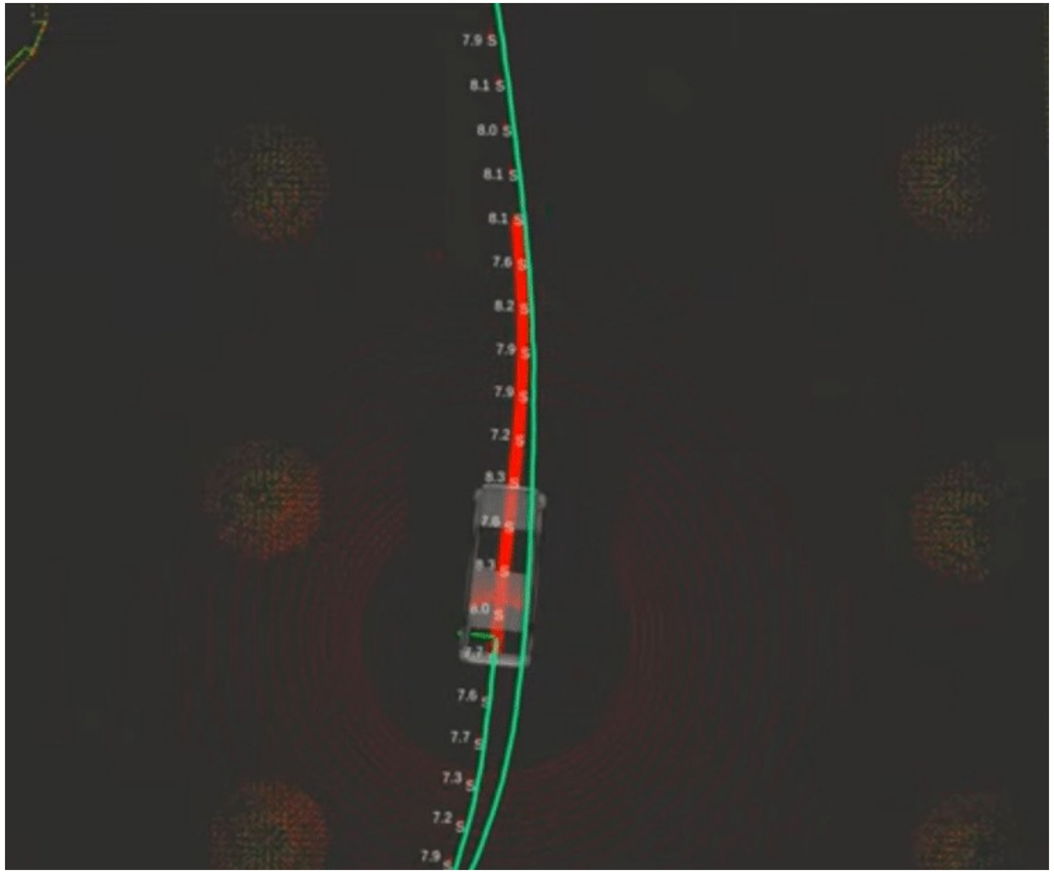
Perception: Localization and Mapping

- Localization in respect to:
  - The environment
  - Objects of interest
- Simultaneous Localization and Mapping (SLAM)
  - Several algorithm based on LIDAR measurements
- CNN
  - Pose estimation
  - Visual Inertial Odometry
- Sensors:
  - LIDAR
  - Depth Camera
  - IMU/Odometry



Planning and Control

- Planning
  - Classical Motion Planning
    - Time-optimal motion primitive (closed-form)
    - Search-based (like Dijkstra)
    - Sampling-based
- Control
  - Closed-loop solutions
    - Proportional–Integral–Derivative controller (PID)
    - Model Predictive Control (MPD)
- Reactive methods
  - Follow The Gap (FTG)
  - Wall following (obstacle avoidance)
- Deep Learning, Reinforcement Learning



Classical Motion Planning and Control

- You can find the theoretical optimal solution
  - If you have enough knowledge about the environment
- Human understandable
- Verificable

- Slow
  - It may require hours
  - You need to simplify the model dynamics to use it in real-time
    - If you simplify the model, the result will diverge from the expected one
    - The higher the speed, the higher the divergence from the real setting

Learning-Based Approaches

- You can use them end-to-end
- Could be more robust
- Faster

Supervised Learning

- Based on imitation
- Current approach by major car manufacturer

Drawbacks

- Training data
  - huge amounts of labelled data or human effort
- Covering all possible driving scenarios is very hard

Learning-Based Approaches

Reinforcement Learning

- Learns by interacting with the environment through trial-and-error
  - ➢ Does not require explicit supervision from humans
- RL is specifically formulated to handle the agent-environment interaction
  - ➢ Natural approach for learning robotics (and autonomous driving)

- Mainly used on simulated environment
  - to avoid consequences in real-life
  - transferring learning from simulations to the real world is a hard problem
    - simulated and real data have not the same distribution
    - agents trained in a synthesized world often fail to generalize
      - if appropriate domain adaptation measures are not taken.

Next Lessons Overview

- We start with the backbone of the platform: ROS
- We will then just introduce some topics from the applicative PoV
  - According to the projects of interest

- Research in Autonomous Driving/Racing has many topics
  - We will mainly focus on high-level software components
  - Topics are just introduced to support the project
  - You will need to deepen the topic you will chose

Lab preparation

- We will use Ubuntu 18.04
- The simulator is lightweight, so
  - You can use your laptop
  - VM is fine

- Install ROS Melodic desktop-full
- Install Tensorflow
  - `$ pip3 install tensorflow`