F1/10 - Autonomous Racing

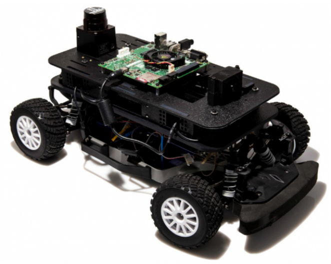
Kuan-Hsun Chen   Junjie Shi

LS 12, TU Dortmund

What is this project group about?

What is this project group about?

- Develop and implement algorithms to finish a racing track **autonomously** and **as fast as possible** using a 1:10 RC car.



Competitions

- Yearly international competition **F1/10**: Student groups compete with each other.
- In parallel: Project-group *F1/10 - Autonomous Racing* at the faculty of Informatik.
- Short-Term: At the end of the first semester there will be a competition against the Informatik team.
- Long-Term: Building a TU Dortmund F1/10 Team.

What are the goals?

Minimal requirements:

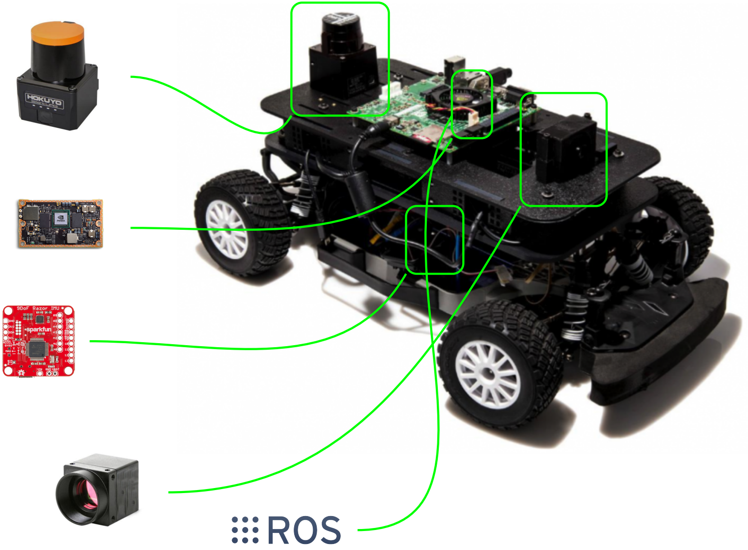
- Fully autonomous finishing of the racetrack.

Wishful:

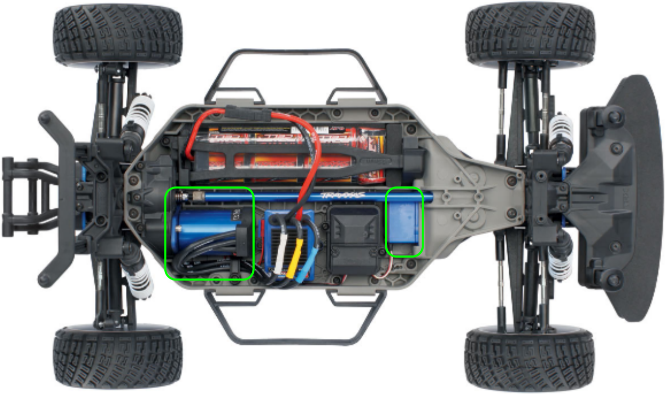
- Algorithms for aggressive maneuvering.
- Overtaking, optimal trajectories, controlled drifting...
- JetsonTX2 hosts a GPU with 256 NVIDIA CUDA cores $\Longrightarrow$ Deep Learning techniques e.g., Convolutional Neural Network (CNN) or complex computer vision algorithms are computationally feasible.

You choose the concepts and algorithms!

Sensing & Perception



Actuation



Actuation

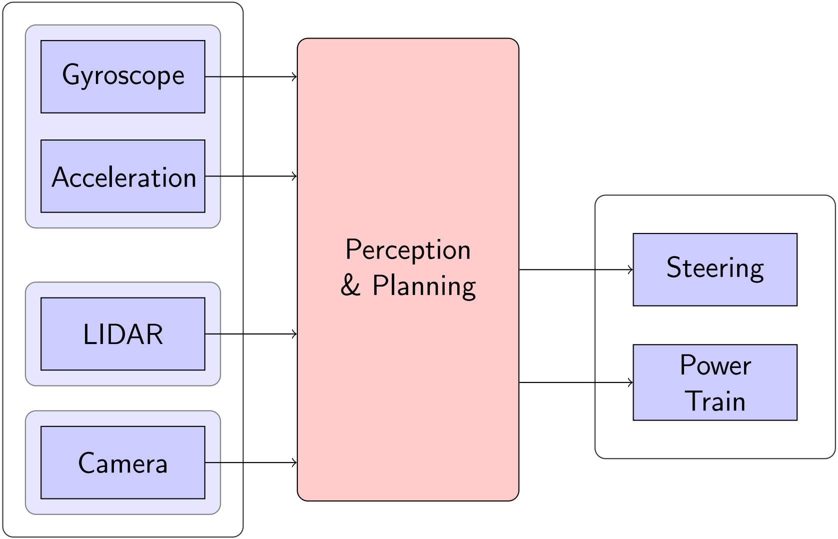
Servo (steering)

- Torque: $9kg \cdot cm$
- Speed: $0.17sec/60°$

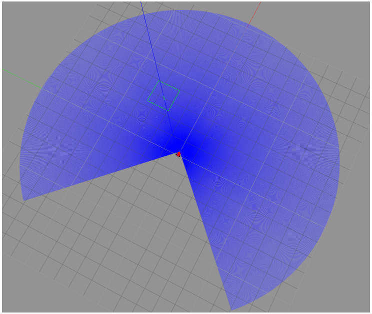
BLDC (powertrain)

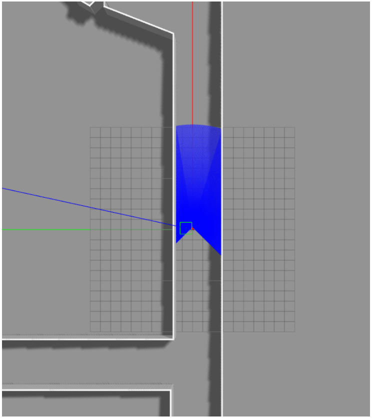
- RPM/volt: 3500 (10-turn)
- Max RPM: 50,000
- Current: 65A constant / 100A peak/burst

System Architecture



Localization & Mapping

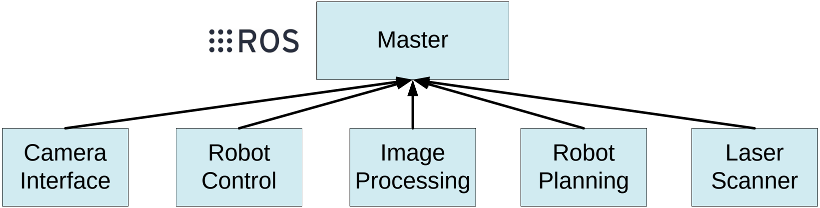




Localization & Mapping

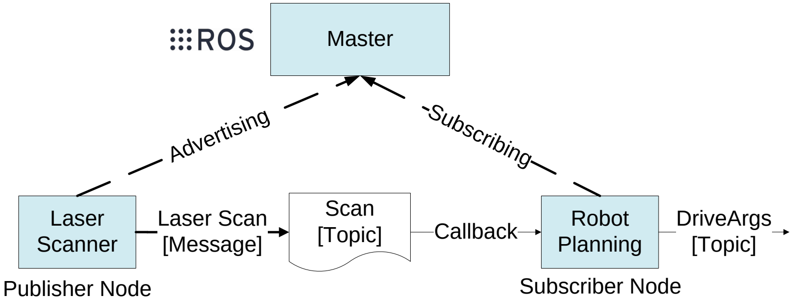
- **Map** of the environment is measured by the car's sensors $\Longrightarrow$ **relative to** the car's **pose**.
- Map depends on the car's pose and the car's pose depends on the knowledge about the car's localization on the map.
- Is considered a **fundamental problem** for autonomous robots.
- SLAM (simultaneous localization and mapping) is a technique for creating a map of the environment and determining the robot's position at the same time.
- Many existing different SLAM approaches.

ROS: Robot Operating System



- It is a distributed framework of **Nodes**.
- Each node runs as a single process individually.
- Nodes in ROS can be written by using C++ Python or (Java).
- Communication with **topics** and **messages**.

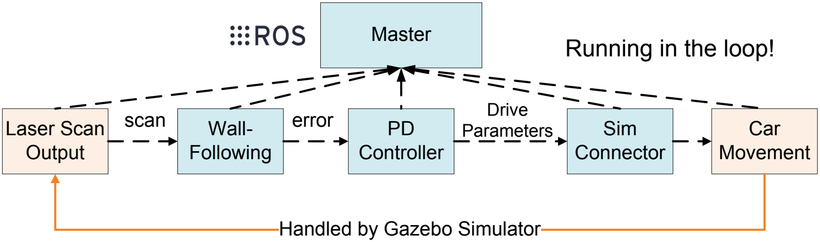
ROS: Robot Operating System



- **Topics** are channels over which *nodes* exchange *messages*.
- **Messages** are the data structure for a *topic*.
  - float32 angle_min, angle_max, scan_time, range_min, etc.

Gazebo Simulator

- Gazebo simulates robots with physics engine, a suite of sensors, and graphical interfaces.



Drifting Control

Model Predictive Control on drifting models[1]

`https://www.youtube.com/watch?v=1AR2-OHCxsQ`

[1] Thanks to Alexander Puzicha for providing this link

Schedules and organizationals

Estimated schedule for the next semester

Task	Weeks
Introductory Phase Seminar Projectplan Tutorials and Problem Identification	3
Analysis Phase Physical modeling of the RC-car Identify minimal set of functionality for autonoums driving Identify problems, usable approaches and algorithms Analyse race tracks and race relevant objectives	4
Concept and Design Phase e.g., Control and Trajectory planning SLAM Sensor processing and Fusion ...	6
Validationphase Demonstration and mock race	2
Documentation and Presentation	1

What should you know?

- Knowledgeable of Python programming.
- Knowledgeable of Engineering mathematics, statistics and probability theory.
- Comfortable with using LINUX.
- Motivation to learn new topics.

What can you expect to learn?:

- How to work a multi-disciplinary engineering problem.
- Robotics, Filters, Estimators, Control, Computer Vision, GPU programming, ...
- Teamwork, Projectmanagement, Softwaredevelopment in a larger team and using version control.
- How to transfer an engineering problem from theory to practice.

Who should apply?

- Multiple subproblems $\implies$ multiple specialties required!
- We are looking for self-motivated students that have a **background or interest** in:
  1. Robotics & Control
  2. Machine Learning
  3. Computer Vision
  4. Softwareengineering, Programming, GPU programming
- So please tell us in the application what your **interests** are!

What is next?

- 06.07.2018 Closing of registration.
- 13.07.2018 Notification of accepted students.
- 20.07.2018 End of lectures SoSe 2018.
- 20.07.2018 First meeting for the selection of seminar topics.
- 01.10.2018 WiSe 2018/19 begins.
- Blockseminar 27.09.2018 - 02.10.2018 (Choose 2 days).

Questions?