Autonomous Racing Car for Formula Student Driverless

Juraj Kabzan, Miguel De La Iglesia Valls, Huub Hendrikx, Victor Reijgwart, Manuel Dangel, Fabio Meier, Ueli Graf, Efimia Panagiotaki
What to expect

1. AMZ Team
2. The Competition
3. Algorithms
4. Software Tools
AMZ RACING

• AMZ overall
  • Since 2007
  • Overall 1st on World Ranking in 2013, ’14, ’15
  • 0-100km/h in 1.513s
• Driverless Team
  • 8 ETH MSc students
  • Helped by 8 AMZ Alumni
• Goals
  • Finish all disciplines
  • Max overall points
  • Pass on knowledge
What to expect

1. AMZ Team
2. The Competition
3. Algorithms
4. Software Tools
Formula Student Germany 2017
FSG Competition

Dynamic Disciplines

- Skidpad: 75 Points
- Acceleration: 75 Points
- Trackdrive: 250 Points
- Efficiency: 100 Points

Static Disciplines

- Business Plan: 75 Points
- Cost: 100 Points
- Engineering Design: 150 Points
- Autonomous Design: 175 Points
Trackdrive and Trackwalk

- Yellow/Blue Cone
- Small/Big Orange Cone
- Red TK Marking & TK Equipment (Shape undefined)

10 Laps

Stop Area (after 10 laps)

6 m

Start Position

3.5 m min.

High Contrast Track Limit Line

3.5 m max.
Skidpad
- Build for Formula Student Electric 2015
- 4WD electric
  - Torque Vectoring
  - Breaking by Recuperation
- Full aerodynamic package
- High wheel torque
- Lightweight
What to expect

1. AMZ Team
2. The Competition
3. Algorithms
4. Software Tools
Overall Concept

Slave ROS
- VSLAM
- Visual Inertial Odometry
- Visual Cone Detection

Master ROS
- Semantic SLAM
- Lidar Cone Detection

Think
- Sensor Fusion
- Fault Detection
- Trajectory generation
- Vehicle Dynamics Control

See
- Low Level State Machine
- Stereo camera + IMU
- LIDAR
- INS
- Odorn
- GSS

Act
- Steering
- Motors
- EBS

ECU (Real Time)
Automatic Launching

- Every discipline is launched from single launch file through `high_level_safety` package
  - Launches all required nodes
  - Monitors used resources
- Monitors heartbeats of every node and stops the car if too many are missing

![Diagram showing vehicle control unit, emergency brake system, and heartbeat packages.]
Control Station and Diagnostics
Slave: Visual Pipeline

- Stereo camera (648 x 488) with INS
- Fusion of ROVIO[1] with ORB_SLAM[2] is open source with our dataset
- Cone detection is based on Cascade Classifier

---


Master: Semantic SLAM (Fast SLAM [1])

- Two main ROS package
  - Simulator, Configurator & Observer tool
  - SLAM/Localization Node (runs on car)
- Efficient workflow for testing
  - Importance of GUI tools

---

SLAM Simulator
Master: Sensorfusion

Pose in the Map

SLAM

INS

Velocity in Car frame

Wheel Odometry

EKF

Outlier rejection

State
EKF Debugging

- Custom debug messages
- Extensive use of RQT Multiplot
- Use of RVIZ Plugins
  - IMU
  - Variance Plotting
- Gazebo Simulation
Two Modes

- Discovery mode:
  - No, or very little knowledge about the track
  - Very slow and map

- Race Mode:
  - Map must be available
  - MPC with dynamic vehicle model
  - Solved online with EMBOTECH Forces Pro at 20Hz
The Switch
Skidpad

- Localize in a predefined Map
- Maximize lateral tire force through a slip angle feedback control
- Faster than a human driver
What to expect

1. AMZ Team
2. The Competition
3. Algorithms
4. Software Tools
Simulation

- Closed Loop Simulation
- Sensors simulation
- Gazebo
- Use of gazebo sensors plugins
- Python model simulating precise vehicle dynamics
Software Tools or “How to Keep it Together”

- Ubuntu 14.04 LTS, ROS Indigo
- Software version control
  - Git
- Extensive use of simulation
- ! Visualizing as much as possible !
Software Tools or “How to Keep it Together”

- Lots of tests on hardware
  - Once a week testing on airports
- Continuous Integration:
  - Jenkins
- Nightly Simulations on a Server
- Logging:
  - Just the Important Information
  - **TAKE CARE:** Sometimes less is eventually more