# Autonomous Driving Challenges

Artur Pereira

artur@ua.pt

University of Aveiro

IRIS Lab

(Inteligent Robotics and Inteligent Systems)

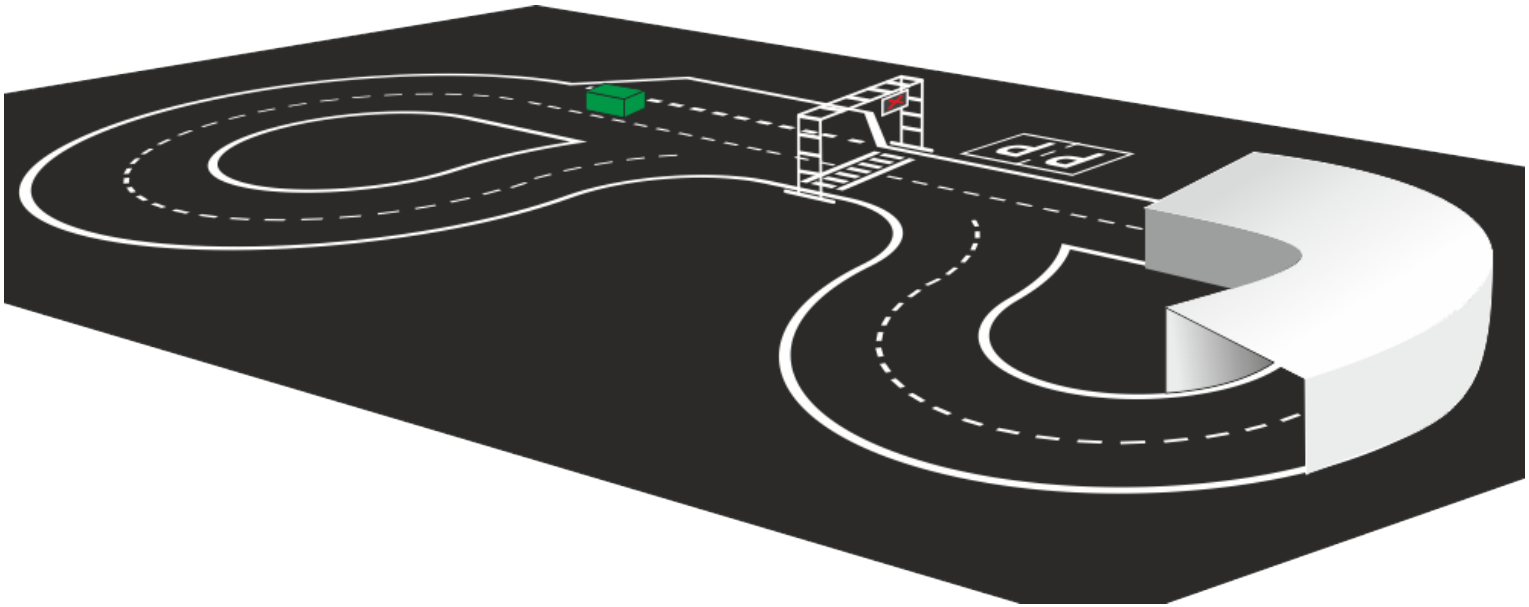
February 20th, 2018

# IRIS Lab

- Robotics Lab, IEETA, University of Aveiro
- Autonomous Mobile Robotics, Intelligent Robotics
  - Robotic soccer
  - Intelligent manipulation
  - Autonomous driving
  - Service Robots
  - Human-robot interaction
  - ...
- Envolved in Robotic Competitions
  - RoboCup (soccer MSL, @Home, 3D simulation league, ...)
  - Portuguese Robotics Open (soccer, **autonomous driving**, rescue, robot@factory, freeBots)
  - MicroRato contest

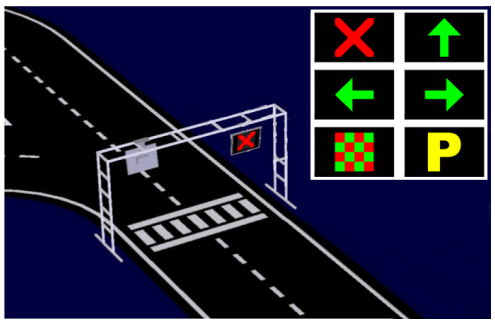
# Autonomous Driving Competition

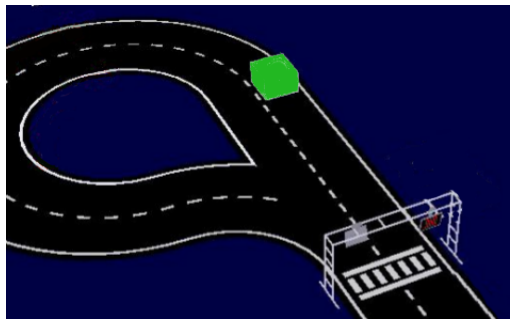
- Goal: develop an autonomous vehicle that can solve challenges in a scenario that resembles a conventional road
- 3 different type of challenges:
  - Driving challenges
  - Parking challenges
  - Vertical sign detection challenge

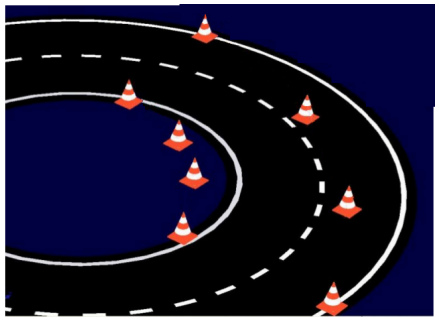


# Autonomous Driving Competition

- Driving challenges: complete 2 laps to the track
  - D1 keeping inside the road and stopping at croswalk
  - D2 D1 plus obeying the signs at the gantry
  - D3 D2 plus avoiding obstacles (and crossing the tunnel)
  - D4 D3 plus driving in a road working area
- Score is based on number of half laps completed, time taken and penalties incurred

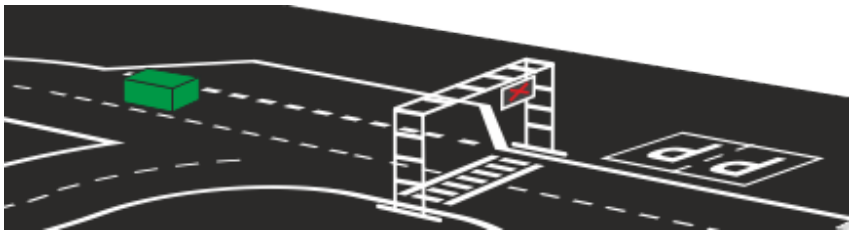


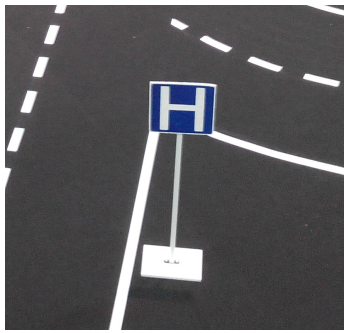


# Autonomous Driving Competition

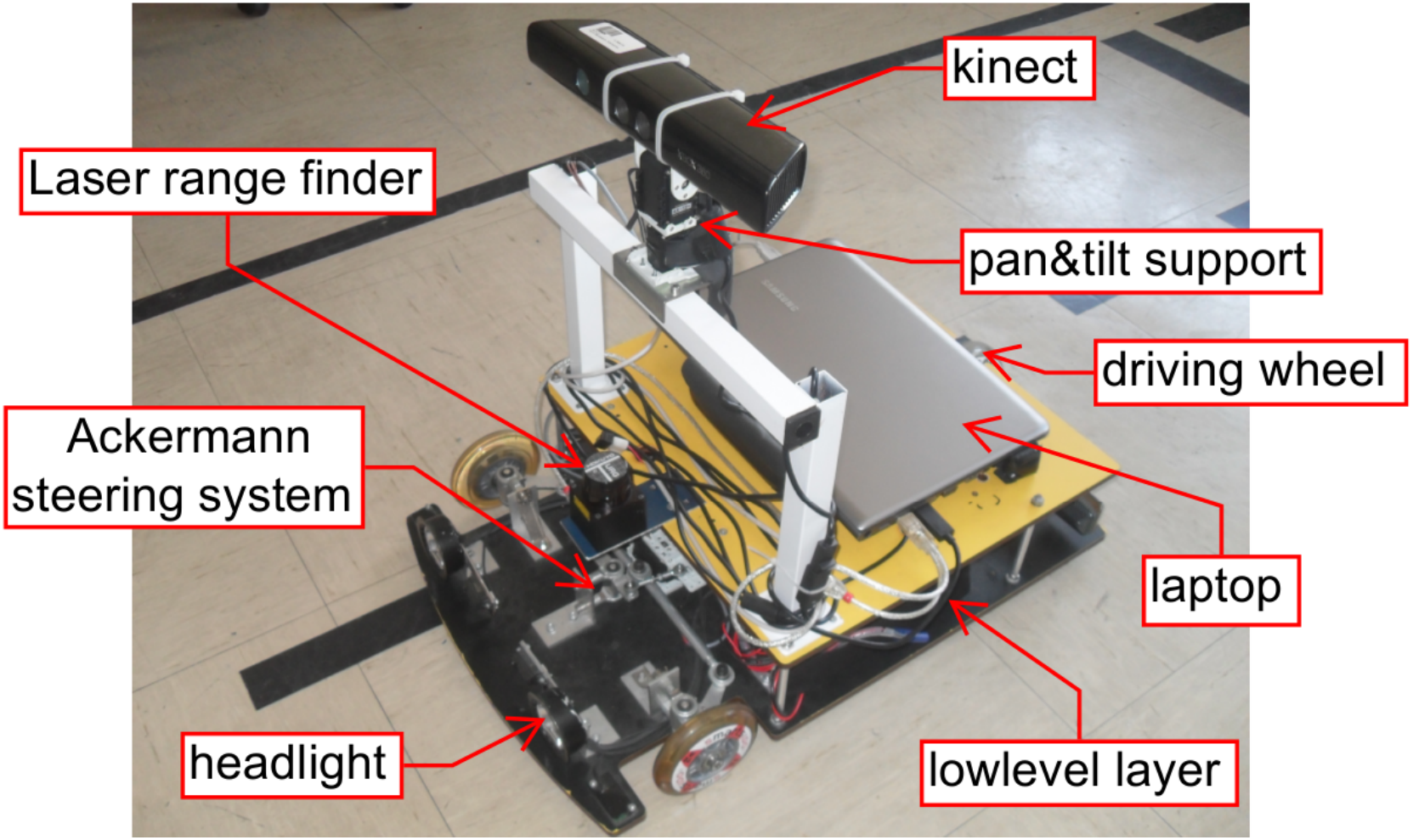
- Parking challenges: parking at the parallel lane or the bay area
  - P1 Parallel lane, without obstacles
  - P2 Parallel lane, with obstacles
  - B1 Bay area, both empty
  - B2 Bay area, one occupied



- Vertical sign detection challenge
  - V1 Identify 6 out of 12 vertical signs, 2 of each type

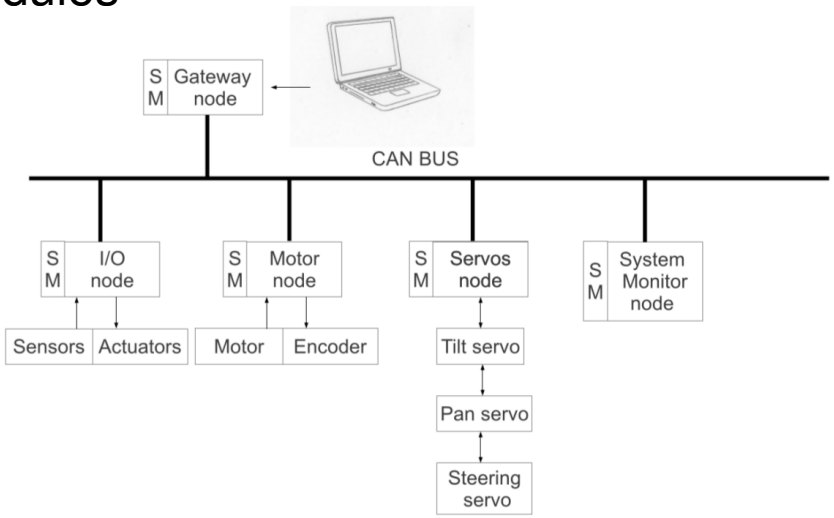



# The ROTA hardware



# The ROTA software

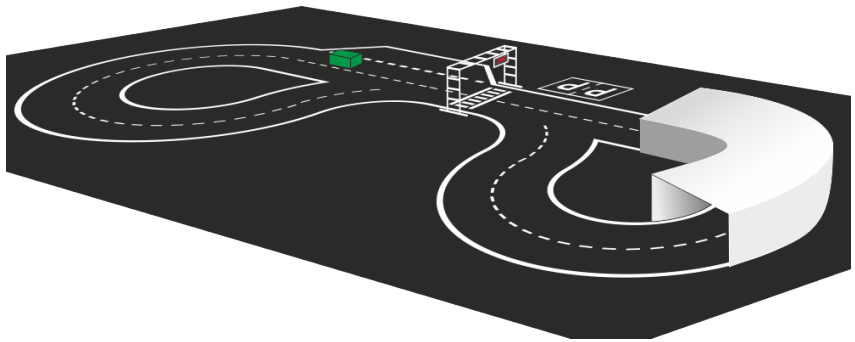
- Low level control layer
  - Based on microcontroller modules
  - Connected to the high level through a serial connection




- High level control layer
  - Running in a laptop
  - Developed on top of ROS (Robot Operating System)
- Realistic simulation environment
  - Supported on Gazebo and ROS

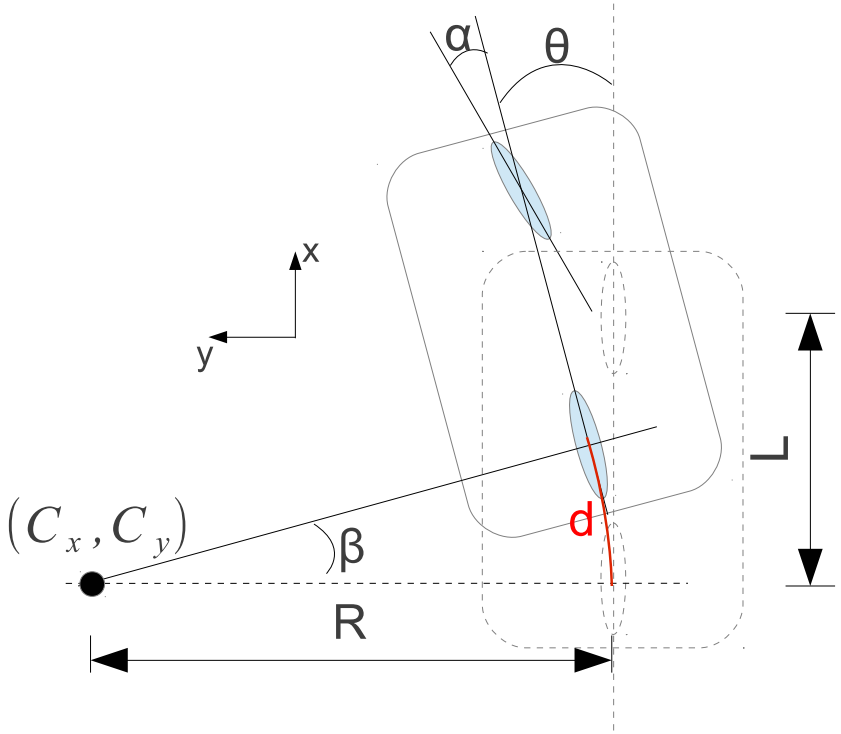
# High level control software

- Modular approach, based on a publisher-subscriber paradigm
- Current existing modules:
  - Low level interface and dead reackoning
  - Image acquisition (color and depth)
  - LRF acquisition
  - Image processing
  - Lane pose estimation
  - World (track) pose estimation
  - Obstacle detection and localization
  - Navigation
  - Gantry sign identification
  - Vertical sign identification
  - Decision (strategy)
  - Mapping
  - Remote control



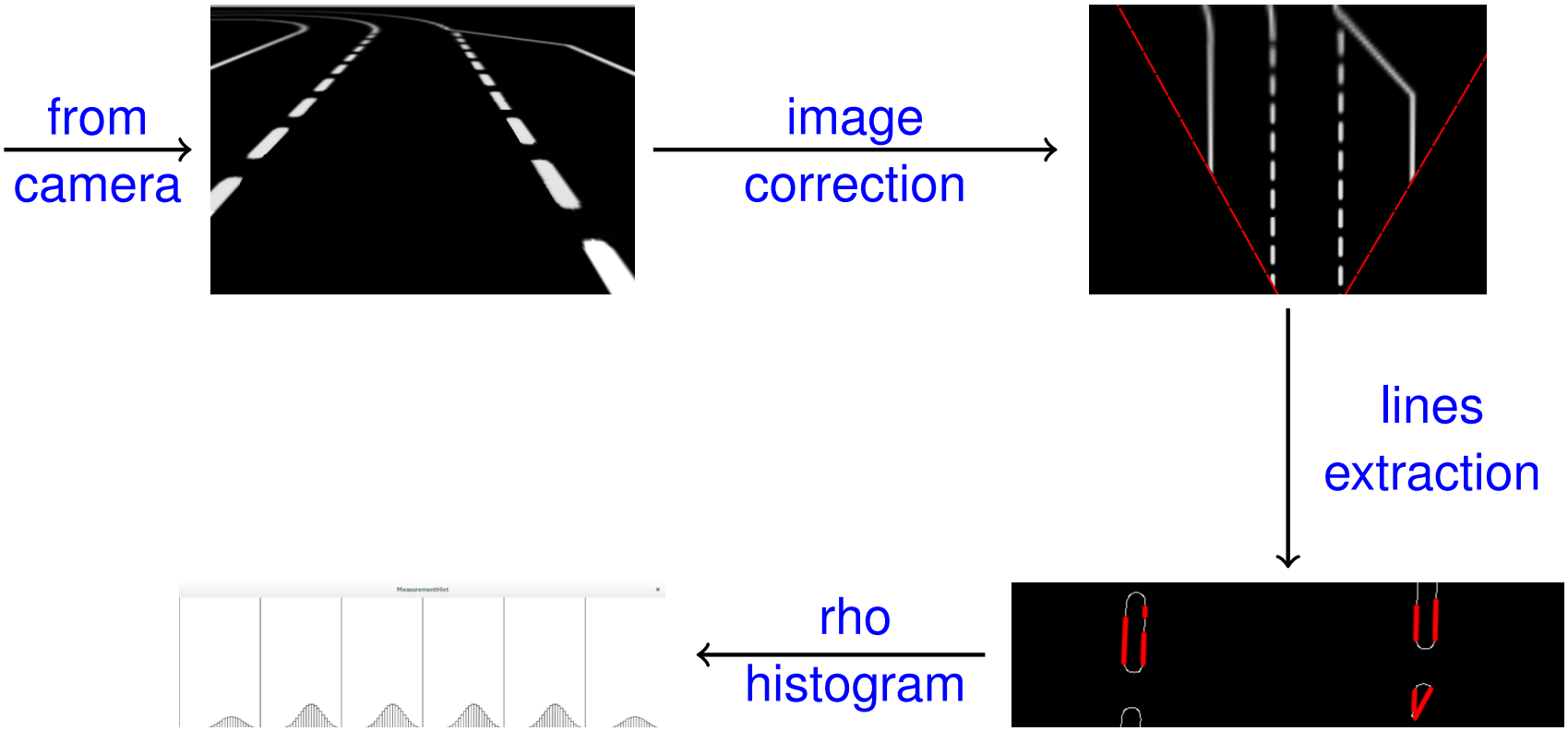
# Dead reackoning

- Based on driving wheel odometry data and steering wheels angular position, update estimation of vehicle's pose
- A virtual wheel, in the middle, between the two real ones, is used in the calculation

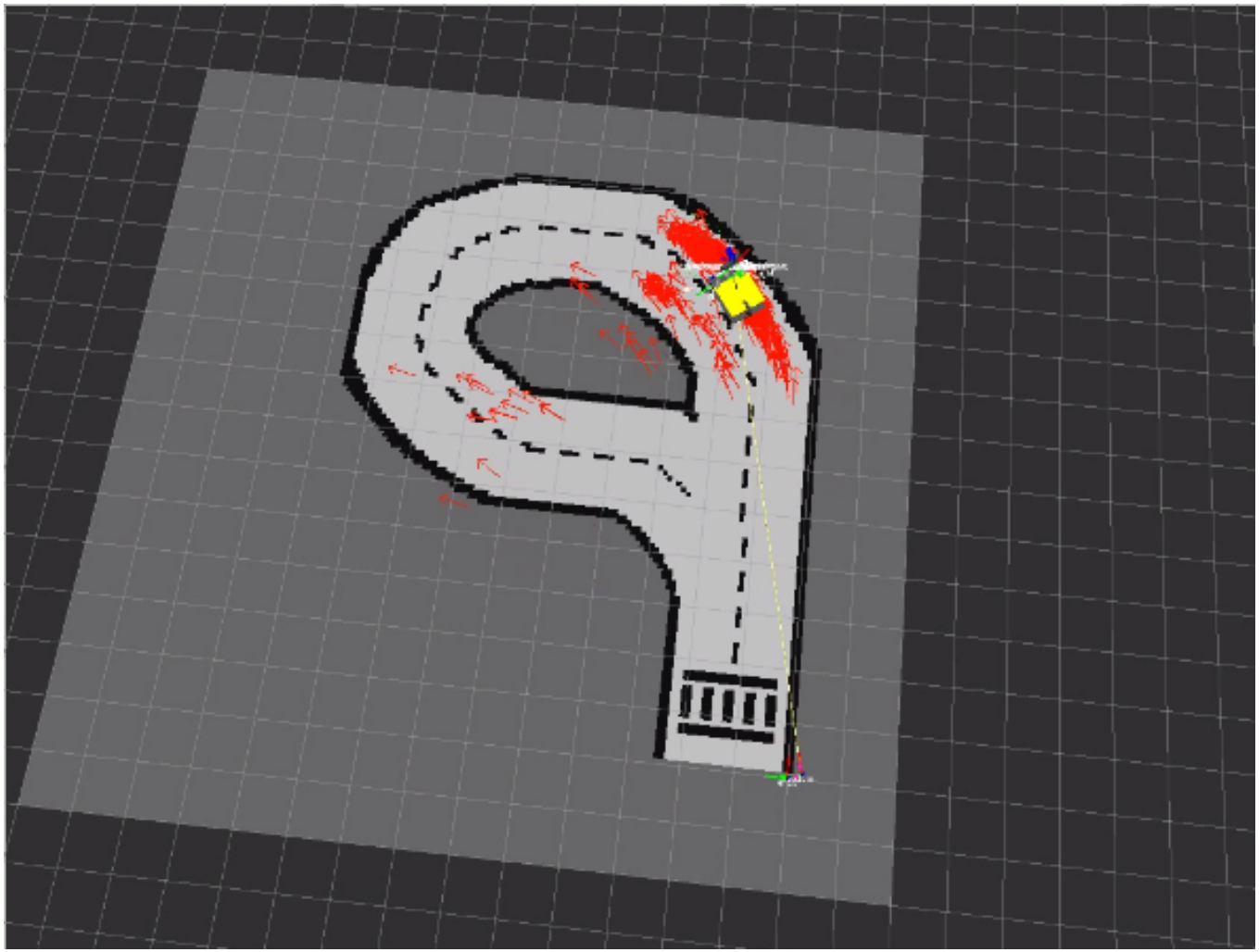


# Lane pose estimation

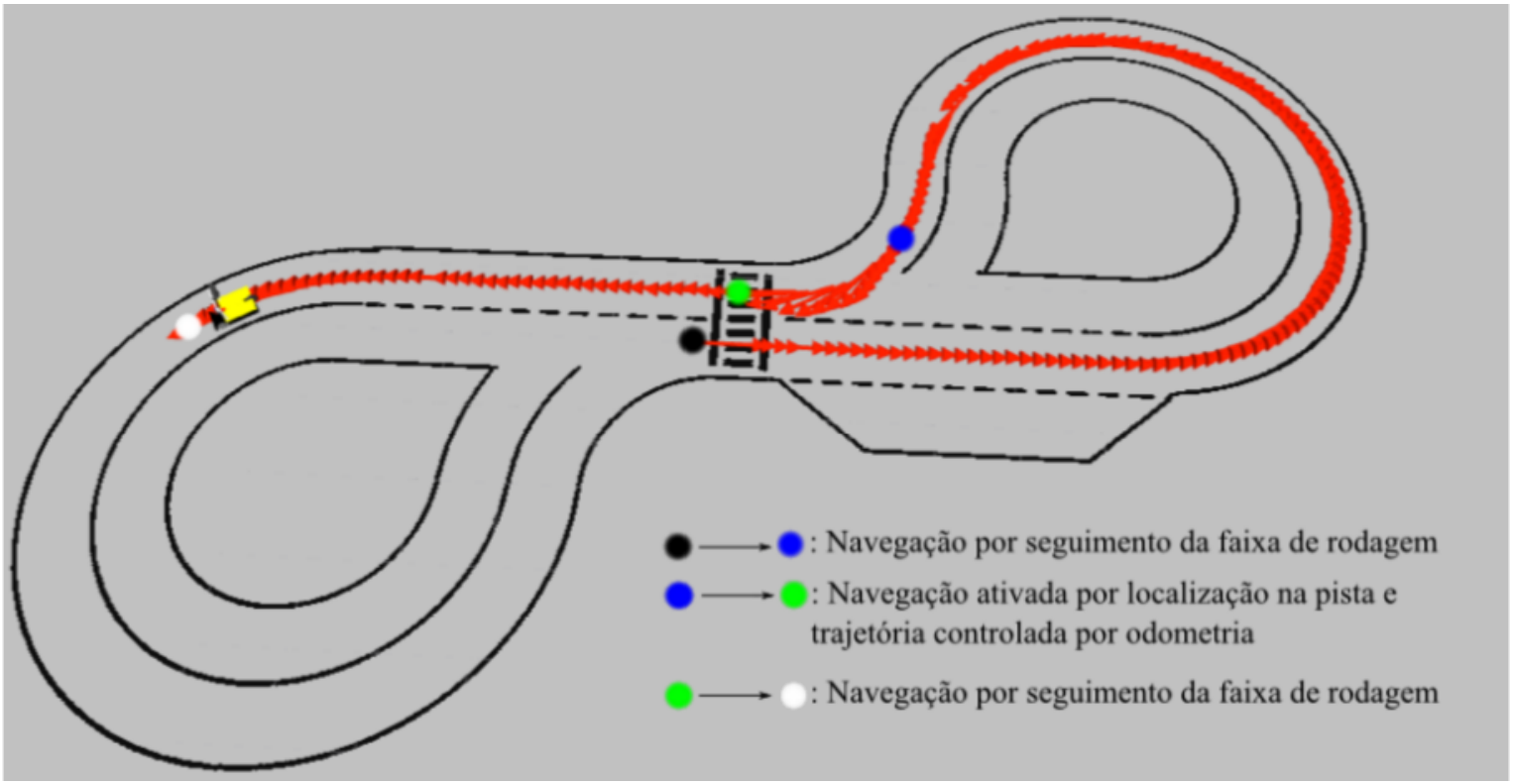
- Lane pose is represented by 2 histograms: one for traversal position and one for orientation



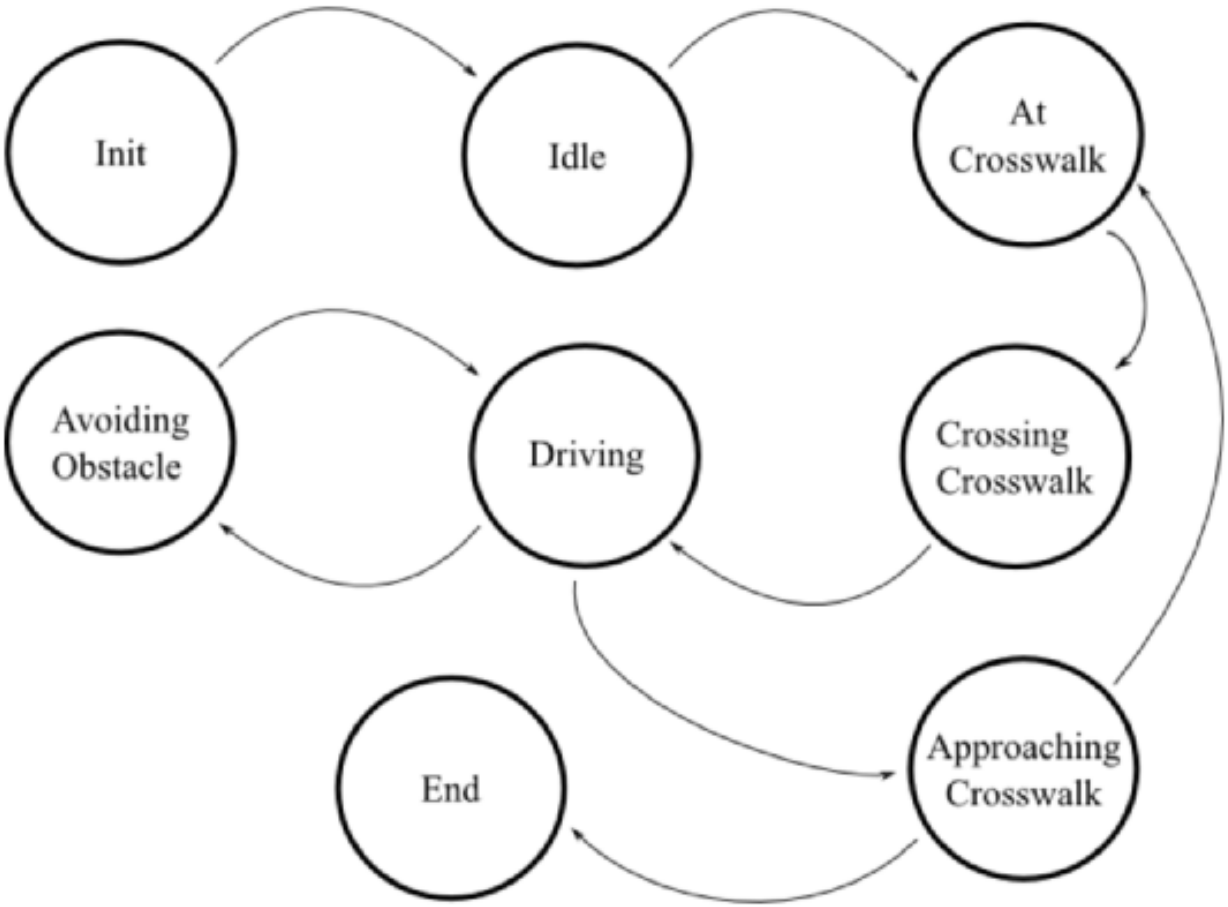
# Particle filter localization



# Hybrid navigation

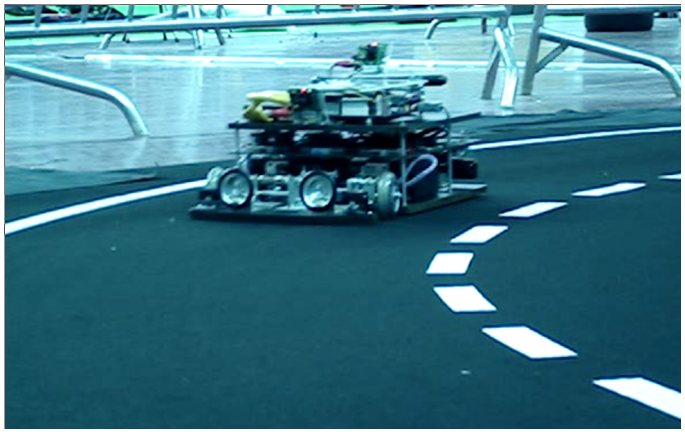


# Decision state machine

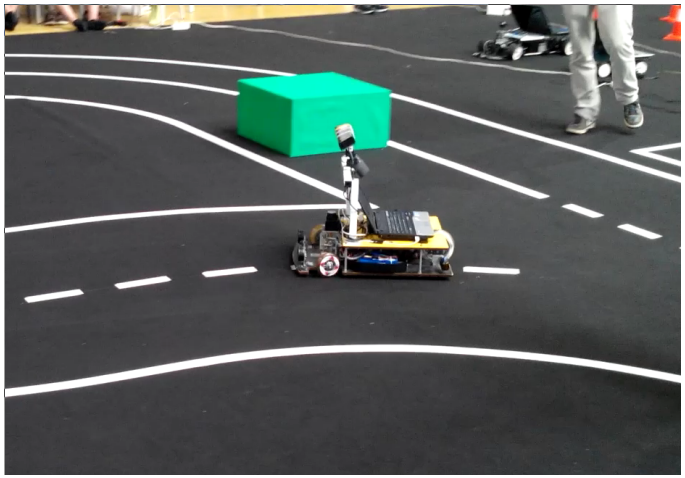


# Two videos

- Guimarães, 2006



- Espinho, 2014



# Building the vehicle