

1. Assignment, Introduction to Robotics WS17/18

Prof. Daniel Göhring,
Institut für Informatik, Freie Universität Berlin
Submission: online until Tuesday, 31 Oct 2017, 11:55 a.m.

Please summarize your results (images and descriptions) in a pdf-document and name it, e.g., "RO-01-<surnames of the students - group name>.pdf".

There should not be any source code in the pdf document.

If you submit source code, use separate files containing the code.

Only one member of the group must submit the document.

Do not copy solutions to other groups.

Every group shall contain two people.

By the end of this class you will need 60% of points to be allowed to write the final exam.

1. ROS Installation (4 Points)

Install ROS kinetic (**requires Ubuntu 16.04 LTS 64 bit Desktop**).

<http://wiki.ros.org/kinetic/Installation/Ubuntu>

Setup your environment variables as described in the above installation tutorial.

Open a terminal window and run the command: **roscore**

Create a screenshot of your console running roscore.

2. Gazebo and model car (3 Points)

Install gazebo with the model car simulator.

Create a screenshot from the simulation.

3. ROS topics and Moving the Simulated Car (3 Points)

a) Print all the topics which are published and which start with /camera
Use the rostopic list command.

b) Publish a topic on the console, e.g. for speed:

```
rostopic pub /manual_control/speed std_msgs/Int16 "data: -10"
```

Or for steering

```
rostopic pub /manual_control/steering std_msgs/Int16 "data: -8"
```

c) Create a screenshot from the simulated camera
after you moved the car.

Copy the topic list from 3 a) and the image from 3 c) into your submission document.

Cookbook - How to install and test the simulator

seat_car_simulator

Developed at ****IRI****, Institut de Robòtica i Informàtica Industrial, CSIC-UPC:
www.iri.upc.edu

Provides a Gazebo simulation of a scaled 1:10 vehicle:
<https://github.com/AutoModelCar/AutoModelCarWiki/wiki>

Used in the SEAT Autonomous Driving Challenge:
<http://www.autonomousdrivingchallenge.com>

Installing

1. Clone AutoModelCar version-3 repo, because we need its odometry package

...

```
mkdir -p ~/seat
```

```
cd ~/seat
```

```
#git clone -b version-3 https://github.com/AutoModelCar/model_car.git model_car_3
```

```
git clone -b version-3-kinetic https://github.com/AutoModelCar/model_car.git model_car_3
```

```
#source /opt/ros/indigo/setup.bash
```

```
source /opt/ros/kinetic/setup.bash
```

```
cd ~/seat/model_car_3/catkin_ws
```

```
rm -rf build devel
```

```
# We actually only need the odometry package
```

```
catkin build odometry
```

...

2. Create our catkin_ws workspace, overlaying model_car_3/catkin_ws

...

```
source ~/seat/model_car_3/catkin_ws/devel/setup.bash
```

```
mkdir -p ~/seat/catkin_ws/src
```

```
cd ~/seat/catkin_ws
```

```
catkin build
```

```
source ~/seat/catkin_ws/devel/setup.bash
```

```
# You can/should add this last line to you ~/.bashrc
```

...

3. Clone simulator in catkin_ws

...

```
roscd; cd ../src
git clone https://gitlab.iri.upc.edu/seat_adc/seat_car_simulator.git
rosdep install -i --from-paths seat_car_simulator
roscd; cd ..
catkin build seat_car_simulator
...
```

Running

...

```
roslaunch seat_car_gazebo sim.launch
```

Test semaphore

```
rosservice call /semaphore_start1_color_plugin/trigger
```

Test car movement

```
rostopic pub /manual_control/steering std_msgs/Int16 "data: 75"
rostopic pub /manual_control/speed std_msgs/Int16 "data: -750"
```