Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Technical Aspects of Multimodal Systems
Department of Informatics

Project Intelligent Robotics
Assignment #2

On this Assignment, you will make first experiences with a real Robot:

The Turtlebot.

You will get in contact with sensors and their visualization. In a second step you will write a program to move the Turtlebot and use the sensor data to stop it safely before colliding with an obstacle.

Real robots require to be handled with care. Be prepared to rescue the robot from hitting obstacles.

Task 2.1 Starting the Turtle and ROS: You can login to the robot via SSH with the username `projectYEAR` on the hosts `donny`, `leo` and `mikey` from any TAMS pool computer. Replace YEAR with the 4-digit year.

2.1.1: Start the Turtlebot

2.1.2: Start ROS and the nodes for a basic setup to use the mobile base and the camera. Good tutorials are available at the Turtlebot wiki page:

```
http://wiki.ros.org/Robots/TurtleBot
```

The most important package is `turtlebot_bringup`.

2.1.3: Start the visualization tool RViz on your local computer. Try to get a live camera image and drive around manually with the `turtlebot_teleop` node. Visualize the point cloud provided by the Kinect sensor. Use the 3D Visualisation Tutorial to read the data provided by the Kinect sensor:

```
http://wiki.ros.org/turtlebot/Tutorials/indigo/3D%20Visualisation
```

Make sure to correctly export the `ROS_MASTER_URI` on your desktop computer before you start the RViz visualisation tool.

Task 2.2 Kinect Listener: Write a listener node which reads the Kinect data

2.2.1: Find out the identifier and the message type of the Kinect topic.

2.2.2: Look up the ROS tutorial for a simple subscriber. There is one for C/C++ and one for Python. You can choose the language you like. Make sure that you are looking up the right version of ROS. You used this tutorial last time for the publisher.

```
C/C++: http://wiki.ros.org/ROS/Tutorials/WritingPublisherSubscriber(c++)
Python: http://wiki.ros.org/ROS/Tutorials/WritingPublisherSubscriber(python)
```

Task 2.3 Simple Movement: Write a node to move the Turtlebot.

2.3.1: Extend your sensor node to move the Turtle bot.

2.3.2: Test your node on the robot. Make sure that there is enough space around the robot.

Never leave the robot unattended!

2.3.3: Extend your node so that the turtle stops, when you hit a key.

Task 2.4 Detect obstacles: Write a program which moves the TurtleBot and lets it stop in front of a soft obstacle.

2.4.1: Use your code from the last 2 tasks and extract the distance to the next object directly to the front of the Kinect.

2.4.2: Now let the robot move forward until it detects an obstacle closer than 1 m. Use a soft obstacle to test your code.

Task 2.5 Optional: Wall following: Write a program which moves the TurtleBot along a wall to discover the surrounding environment.

Task 2.6 Optional: Mapping: Use gmapping on the Turtlebot with RViz to create a map of the TAMS floor.