

# Introduction to Robotics WS18/19 – Version 1.01

## 5. Assignment: Line detection

### 1. Setting the field

Use white tape to draw a straight lane in the carpet with the sizes shown in Fig. 1, length L of a line should be approx. 1 m.

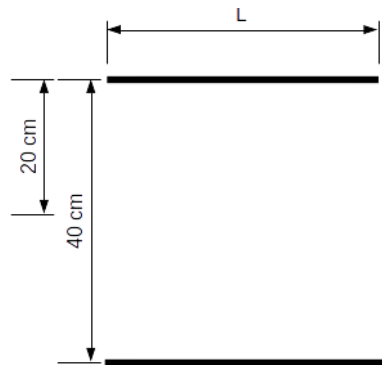


Fig. 1 Lane.

### 2. Lane segmentation (5 Point)

Placing the car in the middle, find a 3d color interval to extract the lines on the road, publish the images on ROS.

You can use the `cv2.inRange` and `cv2.bitwise_and` function to find the colors of your interest on the image: ([http://docs.opencv.org/3.0-beta/doc/py\\_tutorials/py\\_imgproc/py\\_colorspaces/py\\_colorspaces.html](http://docs.opencv.org/3.0-beta/doc/py_tutorials/py_imgproc/py_colorspaces/py_colorspaces.html))

Or you can refer to the code used in Exercise No. 4 last week:

[https://github.com/richrdcm/catkin\\_ws\\_user/tree/master/src/py\\_image\\_processing](https://github.com/richrdcm/catkin_ws_user/tree/master/src/py_image_processing)

### 3. Getting the line equation (5 Points)

Use RANSAC to estimate a linear model ( $y = mx+b$ ) on the image and obtain the two equations of the lines on the road.

Use `cv2.line()` to plot the lines in the original image, publish the resulting image and the (m,b) parameters of each one.