CSE 180
Final Project

Stefano Carpin, PhD
http://robotics.ucmerced.edu
Spring 2018

Logistic Info

- Teams
- Presentation
- Precise description and files available in catcourses (setup mostly similar to lab5 and 6)

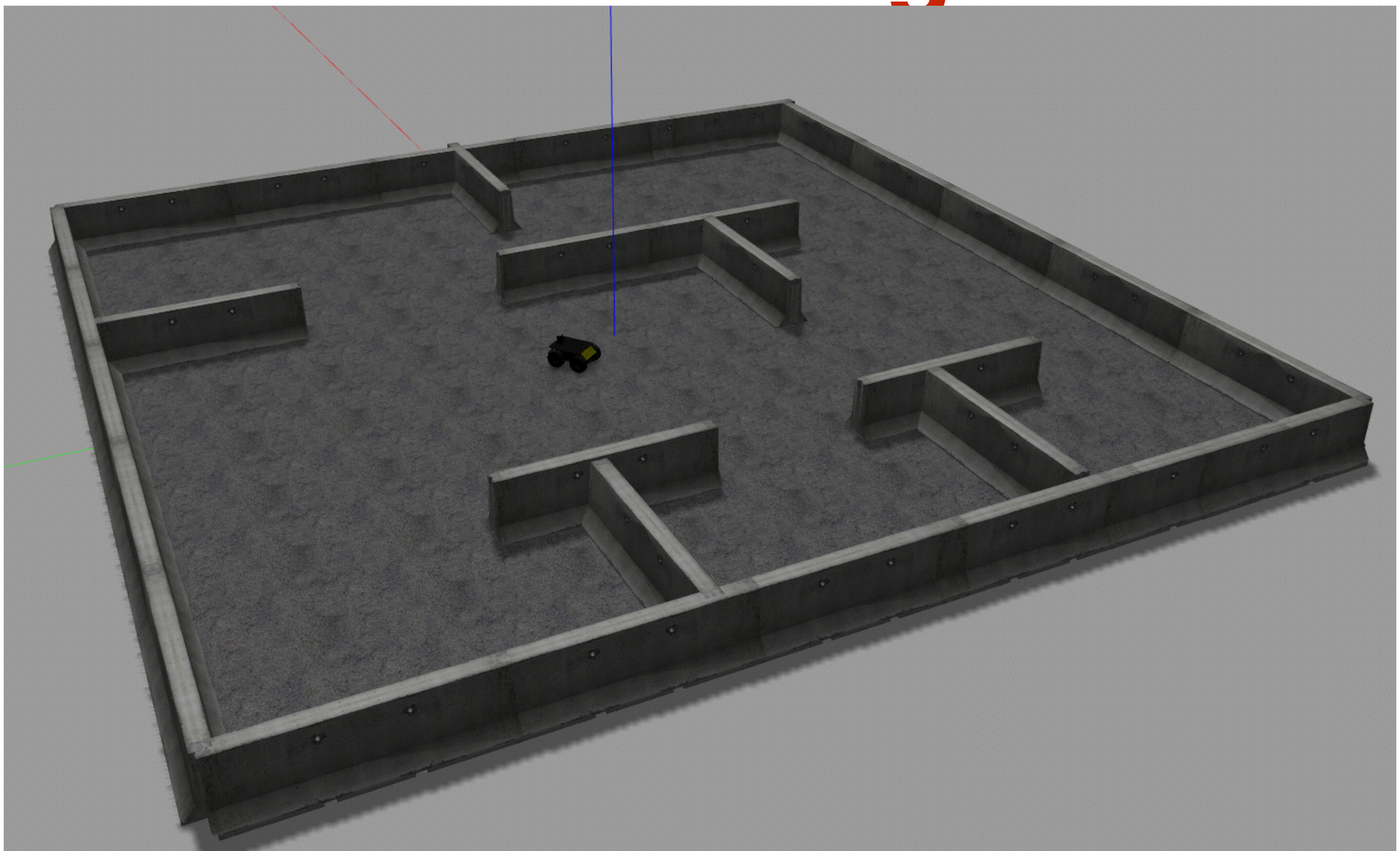
Teams

- Come up with your own team
- No more than 3 individuals per team (no exceptions)
  - individual work allowed but discouraged
- Use Catcourses if you're looking for a team member
- Teams must register on CatCourses by April 2nd (instructions will be posted). If you register you don't get a chance to present (=0 grade)
- One solution per team. All team members get the same grade (no matter what…)

Presentation

- Solution presented during the last week of classes (during scheduled lab time). One team member must present the solution. No show = 0 grade.
- Only registered teams can present
- Presentation consists of
  - showing code running
  - two page report describing the strategy you developed
- Final submission: code (with instructions to run) + report

Robotic scavenger hunt



Problem description

- Given the map of an indoor bounded environment, write a robot controller that finds an unknown number of tags hidden in the environment
- tags are detected by a given sensor (logical camera — more later)
- You must report (print to the screen):
  - how many tags you found
  - for each tag: identity (string), and absolute pose in the map.

Available nodes

- amcl for localization
- move_base to move the robot around
- map_server to get the map of the environment
- logical camera to detect objects in the environment

Map of the Environment

- Can be read from disk or retrieved from map_server
- robot produced. Mostly accurate but with some imprecision
- your code must be generic with respect to the map, i.e., your code will be tested on a map different from the one you are given for testing
- Pseudo info available either from disk (yaml file) or map_server

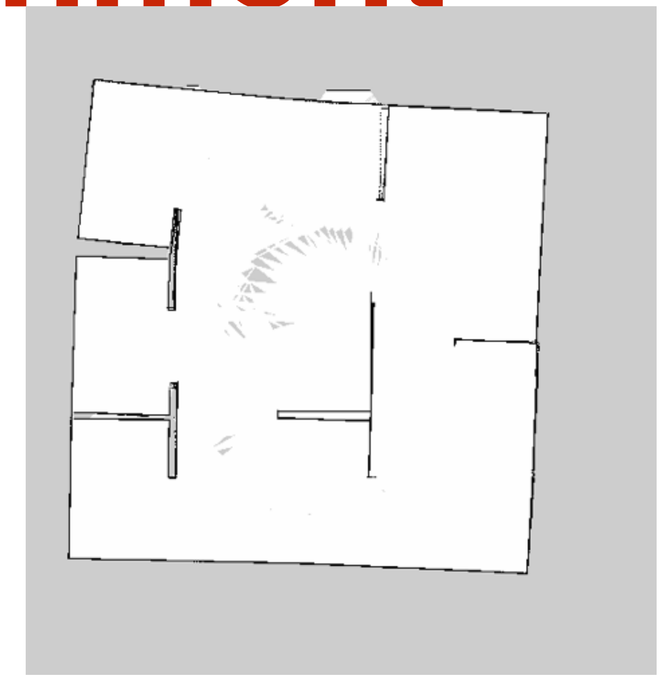
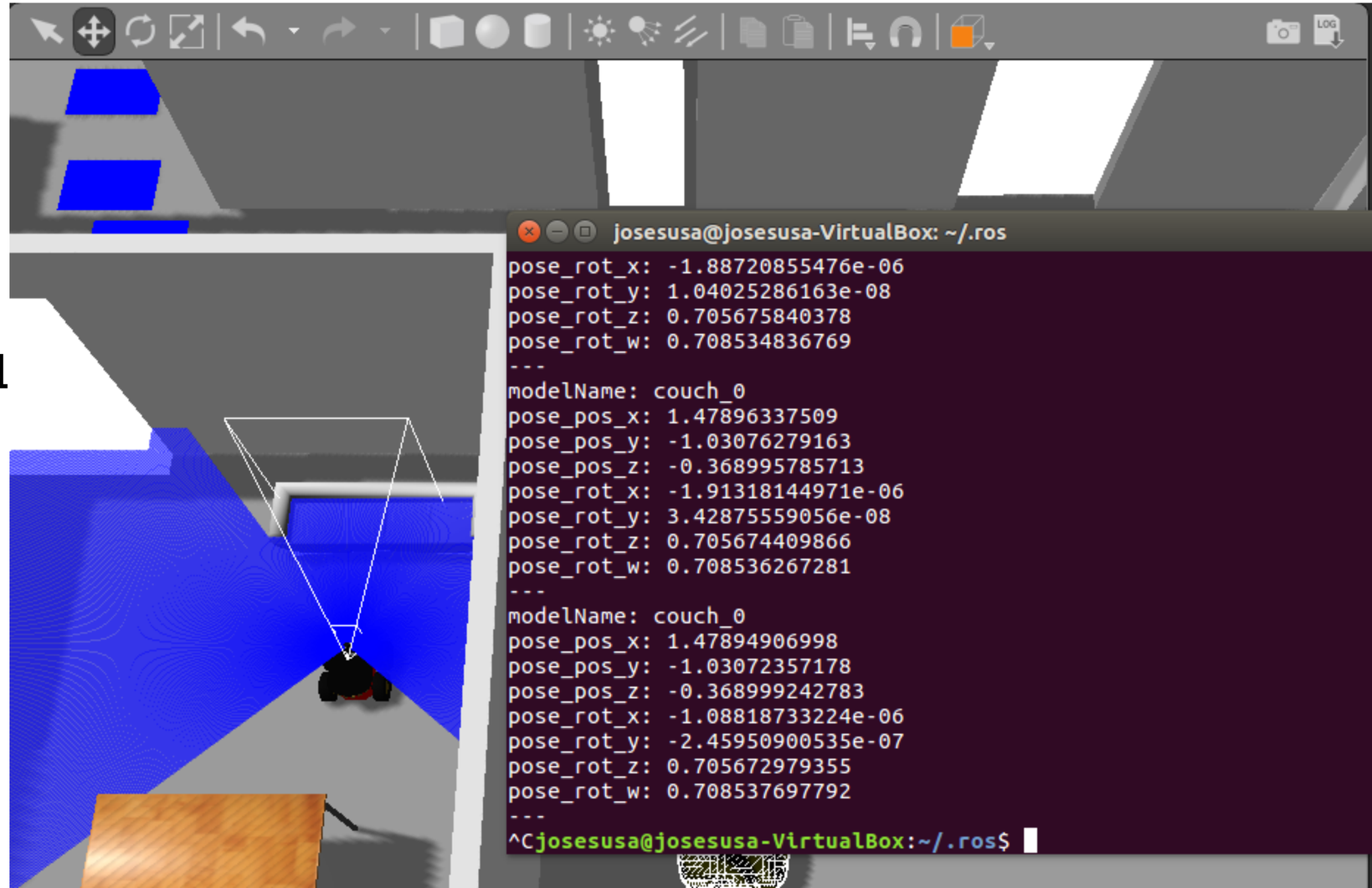


image: anothergoodmap.pgm
resolution: 0.050000
origin: [-100.000000, -100.000000, 0.000000]
negate: 0
occupied_thresh: 0.65
free_thresh: 0.196

Logical camera

- Posts info on **/objectsDetected**



Logical Camera

- publishes messages of type message of type **logical_camera_plugin/logicalImage**

- Example:

```
modelName: couch_0
pose_pos_x: 1.47894906998
pose_pos_y: -1.03072357178
pose_pos_z: -0.368999242783
pose_rot_x: -1.08818733224e-06
pose_rot_y: -2.45950900535e-07
pose_rot_z: 0.705672979355
pose_rot_w: 0.708537697792
```