

# Probabilistic Robotics

PRR06, Fall 2018

## Lab Assignment: FastSLAM with Known Data Association

Assigned: Monday October 8

Due: Sunday October 21; 23:59

A.Visser

October 3, 2018

### Introduction

A Particle Filter is a popular method used for simultaneous localization and mapping, because it is relatively easy to implement, has computational advantages and is robust against data association problems. Applying this to a scenario where only a limited number of landmarks are visible (i.e. six colored landmarks around a soccer field), should be simple.

Yet, this time the algorithm has to be implemented to real robots. The observations of range and bearing are not provided, but should be estimated from the camera images. This means that the landmarks have to be detected, classified and localized in the image under changing lighting-conditions. This location in the image has to be converted to a vector in the local coordination system by including the position of the head and the body of the robot. So, there will be a lot of hard work, frustration and fun to get the particle filter working on the real robot.



Figure 1: The field of the RoboCup Soccer as used in the experiment. The Nao robot will walk several times the 8-shaped path marked on the field.

For this experiment a configuration inspired by the experiment by Steffen Gutmann is used [1]. Steffen directed an Aibo over the field with a joy-stick, and recorded the odometry updates and the observations of the landmarks of the Aibo. It is your task to do the same with a Nao robot. The Nao has a limited field of view, so only 1-2 landmarks are visible at the same time. The clue to detect landmarks is the pink band each landmark has. By looking above and below this band you should be able to reliably classify the signature of the landmark.

Around the field six landmarks are visible, color coded with bands of pink combined with green, yellow and blue (see Fig. ). The green landmarks are placed on locations  $(0,215)$  and  $(0,-215)$  cm. The yellow landmarks are located at  $(315,215)$  and  $(315,-215)$  cm. The blue landmarks are placed at  $(-315,215)$  and  $(-315,-215)$  cm.



Figure 2: The landmarks which are placed around the RoboCup Soccer field.

On the field five black marks are made with black tape. Because the size of the soccer field is 6x4 meter, twice as large as the field used in the Gutmann dataset [1], the marks are placed further apart (see Fig. ). The marks are placed in an eight-shape figure on locations (0,0), (0,100), (100,-100), (-150,0) and (-150,100)cm.

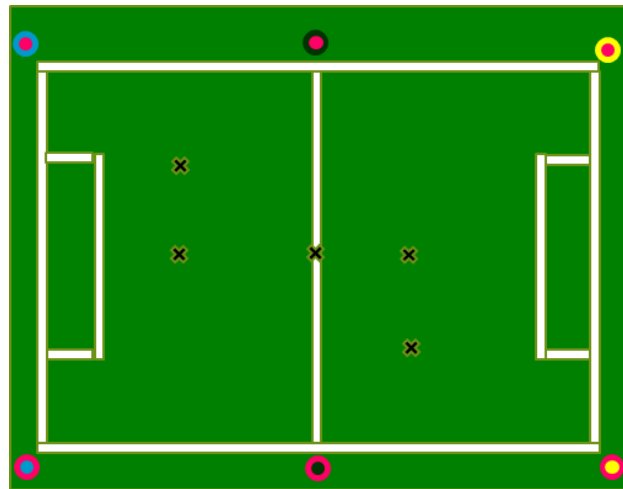


Figure 3: The layout of the RoboCup Soccer field, with the markers on the field and the landmarks around the field.

For your task four Nao humanoid robots with NaoQi are available: Tom, Julia, Carlos, Brooke. The other Nao robots in the lab have an hardware problem or are running the Dutch Nao Team code. A short introduction how to use the robot is available<sup>1</sup>. You can program the robot in Python.

## 1 Task 1 - Data Acquisition

Create a dataset by combining the following Python scripts, which drive the robot around with the keyboard<sup>2</sup> with a script to take pictures from the top camera<sup>3</sup>. Additionally, you need the `naomanager.py` script<sup>4</sup>.

To use a robot from the lab you need one of the operational robots from the RoboLab (C3.165). When you need access to the robolab, contact [A.Visser@uva.nl](mailto:A.Visser@uva.nl) with the following information:

<sup>1</sup>[https://staff.fnwi.uva.nl/a.visser/education/ZSB/Experiment/nao\\_guide.pdf](https://staff.fnwi.uva.nl/a.visser/education/ZSB/Experiment/nao_guide.pdf)

<sup>2</sup>[https://staff.fnwi.uva.nl/a.visser/education/ProbabilisticRobotics/2017/NaoSLAM/keyboard\\_control.py](https://staff.fnwi.uva.nl/a.visser/education/ProbabilisticRobotics/2017/NaoSLAM/keyboard_control.py), courtesy Michiel van der Meer

<sup>3</sup><https://staff.fnwi.uva.nl/a.visser/education/ProbabilisticRobotics/2017/NaoSLAM/datasetCreator.py>, courtesy Douwe van der Wal

<sup>4</sup><https://staff.fnwi.uva.nl/a.visser/education/ProbabilisticRobotics/2017/NaoSLAM/naomanager.py>

- studentnummer
- passnummer

You can turn the robot on by pressing the chest button. It is possible that the robot does not have the Naoqi operating system installed on it. To check if its compatible you can press the chest button after startup. If the robot starts talking it will be Choregraphe compatible, otherwise use a different robot or ask a teaching assistant for help.

If you have a robot ready you can connect it to your computer with an Ethernet cable. If your laptop doesn't have an Ethernet connection you could ask for a Ethernet-USB converter. Press the chest button once. The robot should start to say its IP address. After connecting with the Nao robot you can start programming the Nao in Python. Be carefull when walking with the robot with an Ethernet cable in its head, because a fall could damage its motherboard.

If the Nao robot is not connected with an Ethernet cable, it should say an IP-address starting with 146.50. You could directly access this robot wirelessly when your laptop is connected with the access-point robolab with the password written on the whiteboard. You could check your connection with the command `ping IP-address-as-said`.

## 2 Task 2 - Detect the landmarks

Find the landmarks by searching for objects<sup>5</sup>. Check the size, shape and location above the horizon to be sure that you have a landmark. Check the color above and below the pink band to get the signature of the observation  $s_t^i$ . Estimate the range  $r_t^i$  from the size of the pink object. Estimate the bearing  $\phi_t^i$  by combining the center of the object in the image with the HeadPitch angle of the Nao robot. Try to get statistics on the standard deviation of your observations.

A number of example images are available for testing (see Fig. 2)<sup>6, 7, 8, 9, 10, 11</sup>.

---

<sup>5</sup><https://staff.fnwi.uva.nl/a.visser/education/ProbabilisticRobotics/2017/NaoSLAM/ColorRegion.py>, courtesy Sebastian Thrun

<sup>6</sup><https://staff.fnwi.uva.nl/a.visser/education/ProbabilisticRobotics/2018/NaoSLAM/yp.jpg>

<sup>7</sup><https://staff.fnwi.uva.nl/a.visser/education/ProbabilisticRobotics/2018/NaoSLAM/py.jpg>

<sup>8</sup><https://staff.fnwi.uva.nl/a.visser/education/ProbabilisticRobotics/2018/NaoSLAM/gp.jpg>

<sup>9</sup><https://staff.fnwi.uva.nl/a.visser/education/ProbabilisticRobotics/2018/NaoSLAM/pg.jpg>

<sup>10</sup><https://staff.fnwi.uva.nl/a.visser/education/ProbabilisticRobotics/2018/NaoSLAM/bp.jpg>

<sup>11</sup><https://staff.fnwi.uva.nl/a.visser/education/ProbabilisticRobotics/2018/NaoSLAM/pb.jpg>

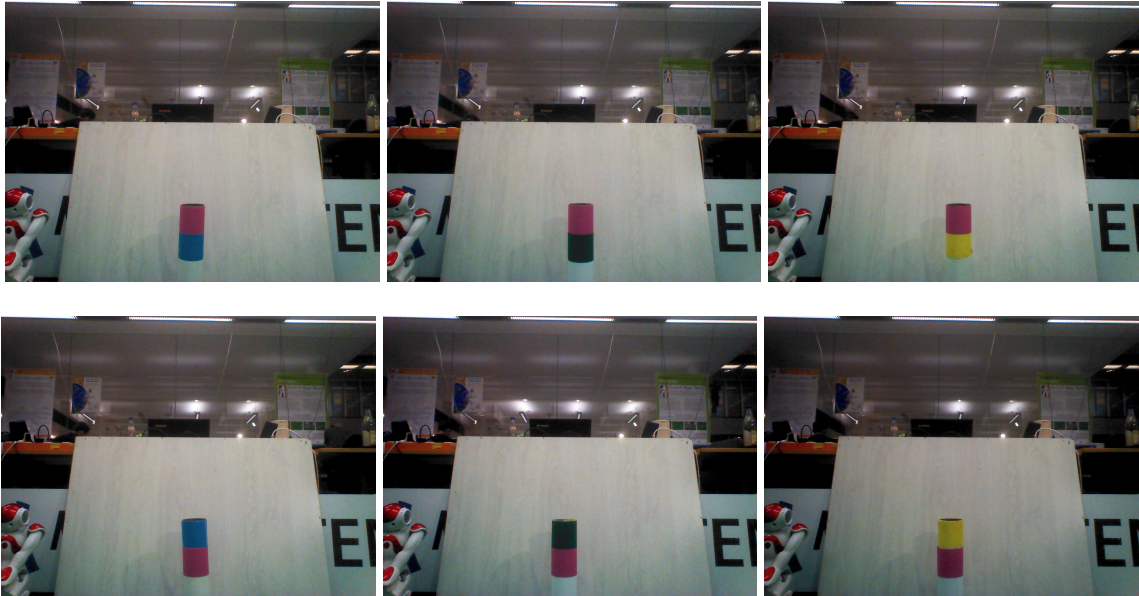


Figure 4: An example of each of the landmarks to test your detection.

### 3 Task 3 - SLAM

Implement the algorithm in Table 13.1 of the book. Analyze the performance and suggest possible improvements.

### 4 Hand-In

You can work on this assignment in with two-pairs (four people). When you have completed the assignment, upload both your solution to Canvas. This should be a PDF, with your python scripts as pseudo-code (for example with the lslisting environment).

### References

- [1] J.-S. Gutmann and D. Fox, “An Experimental Comparison of Localization Methods Continued”, in “Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS’02)”, pp. pp. 454–459, October 2002.