97 p

Robot Design Lab
Tutor: Teena Hassan, Mihaela Popescu

WiSe 2022/2023
Group: 09

Joel Köper
Maarten Behn
Nhu Van Nguyen

Worksheet 03

Hand-in date: November 30, 2022

1 Mapping

Very good

1.1 Quiz on Mapping

15%

3%

1. **What is mapping in robotics? (3%)**

   Map = model of the environment

   Mapping in robotics is a crucial key element in localization performance. The robot constructs/uses a model of the environment(map), therefore mapping the environment. Generally, robotic mapping is the area that deals with the study and application of the ability to localize itself in a map and sometimes to construct the map by the robot.

   The robot explores the environment with the assistance of sensors like laser scanners or cameras for instance. It then collects the measured data over time, fuses them and generates a map of the environment. Noteworthy is that multiple kinds of maps exist in robotics like "grid maps" or "feature maps". Grid maps show if the environment is free or occupied and feature maps mark the position of obstacles and objects in the world.

   Mapping answers the question "How does the world look like?" as a prerequisite for navigation performance.

3%

2. **What is the purpose of loop closure in localization and mapping? (3%)**

   The concept of loop closure is an important key aspect in mapping as well as localization.

   The loop closing is responsible for deciding if the robot has returned to a formerly visited area. A possible option for detecting loop closures is to identify the features that have been already perceived and match them to the currently perceived features. After an implied loop closure is detected by the algorithm, it will merge the two segments and create a more accurate map. Therefore a spotted loop closure updates the previous map and position measurements to correct the drift that is caused by the inaccuracies in odometry or sensor data.

3%

3. **What is the meaning of the following occupancy grid mapping algorithm parameters? (3%)**

   • map resolution (˜delta)

   - It shows the resolution of the model of the environment (map) in metres per occupancy grid block

   • occupancy threshold (˜occ thresh)

   - occ thresh stands for occupancy threshold on gmapping's values. Grid cells with high occupancy are deemed occupied.

   • map update rate (˜map update interval)

   - The period of time in seconds how long it takes to update the map in between updates. The lower the number, the higher the update rate.

4%

4. **Explain the laser scanner inverse sensor model used for occupancy grid mapping. (4%)**

   The inverse sensor model defines cell occupancy. In grid maps, the environment is displayed in a 2D world plane and divided into grid cells, which can be either occupied or free where only free cells can be navigated by the robot. If neither of them is detected then it's a prior (unchanged) cell occupancy. The laser scanner has an optical axis that senses the distance between the sensor and the cell under consideration of Occupancy probability. The laser beam can measure the distance to an object because it detects the reflections that are coming back from the obstacles.

2%

5. **Probabilistic maps: If the robot senses an obstacle 2 meters ahead, why don't we directly enter this information into the map? Why do we need a model based on probability? (2%)**

   We use a model based on probability because the several challenges in robot mapping can have different weights and results. The sensor data can have several sources of errors and some of them are coming from a random dependency that occurs because of the errors inherent in the robot's motion or odometry which also accumulate over time and affect the measurement results. An example could be that the robot is in a place where a lot of dynamic obstacles are present such as a walking person in a store that changes their location. The score for dynamic obstacles would be different in this circumstance. So it is worthwhile to process an already-seen object differently than a not-yep mapped object.

30%

1.2 Occupancy Grid Mapping

1. **Create a map of the maze using the following values for the map resolution parameter delta: 0.01, 0.05, 0.5. Save the three maps. (15%)**

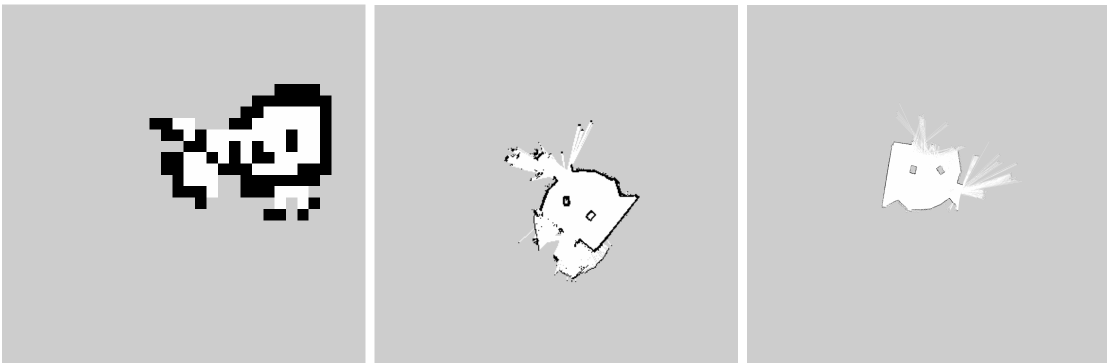


Figure 1: Maps with a resolution delta of 0.5, 0.05 and 0.01

2. **Tune the parameter delta until you obtain a map that looks desirable to you in terms of minimising the chances of hitting obstacles and minimising the storage space. Save this map. (5%)**

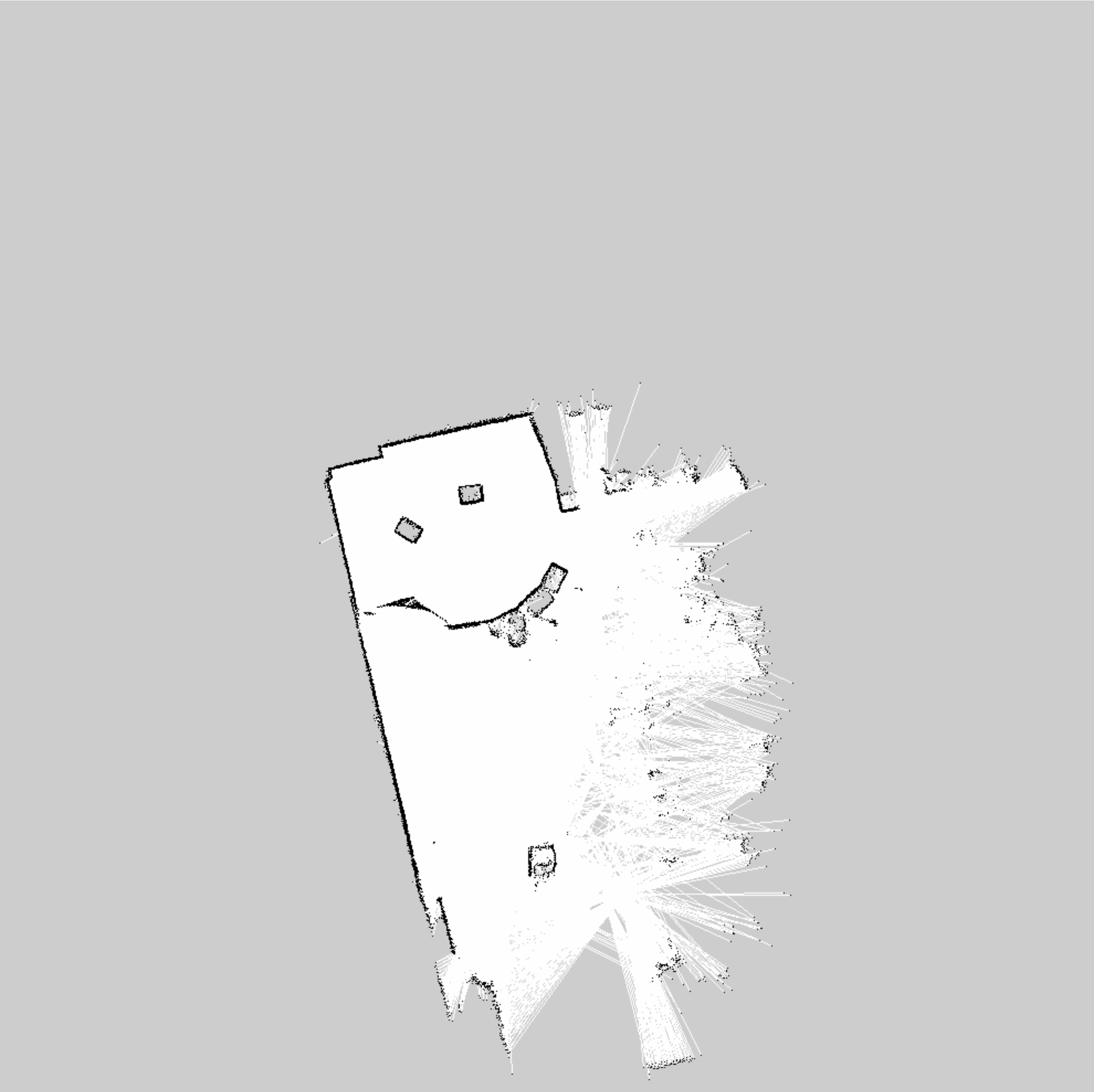


Figure 2: Map with a resolution delta of 0.02

3. **Describe your observations and findings during mapping. Tell which value you found for the delta parameter in subtask 2 and motivate your choice. (10%)**

   We chose a resolution delta of 0.02 because we wanted to create an accurate map, that is still not quite as large as the 0.01 map. Also, the 0.01 had quite a lot of holes in the walls because the laser scanner just doesn't hit every centimetre of the wall. But with a map resolution of 0.02 and careful mapping, we were able to create an accurate map of quite a large area. An 0.02 map is still quite large but we thought that the 0.05 map was just too inaccurate when representing round or very thin objects. Also, we had no performance issues with the 0.02 map therefore there was no real reason to choose a bigger resolution.

The map looks and it seems you also mapped the area outside the maze. For larger surfaces, the size of the map with a resolution of 0.02 might cause storage and computational problems. For instance, occupancy grid maps are used in path planning to find an optimal path from a start cell to a goal cell. The larger the number of cells in the map, the longer the computation of the path planning algorithm.

2 Localization

12% 2.1 Quiz on Localization

2% 1. **What is robot localization? (2%)**
Robot localization refers to the task of computing the pose of a robot relative to a map.

2% 2. **What is the kidnapped robot problem and which strategy can be applied to cope with it? (2%)**
The kidnapped robot problem describes a scenario where the robot is suddenly moved by an external force which results in an offset from the computed robot pose to its actual pose.
This discrepancy can be alleviated in three different ways:

- Introducing variables to control a random addition of potential poses all over the map (Referred to as recovery_alpha_fast and recovery_alpha_slow) which either disappear if incorrect or become a nucleation site for new estimated poses.
This method however is quite slow and can fail by having multiple similar points on the map leading the robot to compute a false pose.
- Guessing the pose of the robot is easy if you can see the robot.
This results in the point cloud of poses being moved to the estimated location, which then has it easy to adapt to the actual pose.
- A last ditch effort, if nothing is working, is to re-initialize the point cloud from zero which has the disadvantage of being slow.

0% 3. **Monte Carlo Localization can work without being adaptive. What does Adaptive in AMCL mean and why is it useful to have an Adaptive Monte Carlo Localization? (3%)**
The *adaptive* part of AMCL is responsible for restricting the tested area to a sensible size by estimating the area of possible poses based on the estimated position calculated from the estimated movement since the last calculation. This results in a more efficient usage of computational resources because it does not have to check the entire map.
The "adaptive" filter decreses the number of particles in the filter when the position certainty is high.

8% 4. **Explain the following sensor model parameters of the AMCL algorithm (8%):**

- measurement noise **(z_hit)**
z_hit is the expected noise around a measurement of an obstacle due to refraction of the laser.
- maximum range **(z_max)**
z_max represents the farthest measurement the robot will accept. This limit is necessary because measurements which exceed the intended range of the sensor are most probably erroneous.
- random measurement **(z_rand)**
z_rand is the expected amount of random measurements which are caused by errors in the sensor or outside interference.
- unexpected obstacles **(z_short)**
z_short are measurements of obstacles which are passing like humans, animals or moving objects. Crosstalk can be another source of such measurements. This variable is called short because those measurements appear close to the sensor.

30%

2.2 Adaptive Monte Carlo Localization

15%

1. **Navigate in the map until all particles converge to a small area. Take screenshots of the particles and the amcl pose visualization in rviz2 at a few seconds interval. Include in your solution 4 screenshots that show the progress of the AMCL algorithm from the initial robot state until the final state. Describe the evolution of the uncertainty ellipses for the robot position and orientation. What do they represent? (15%)**

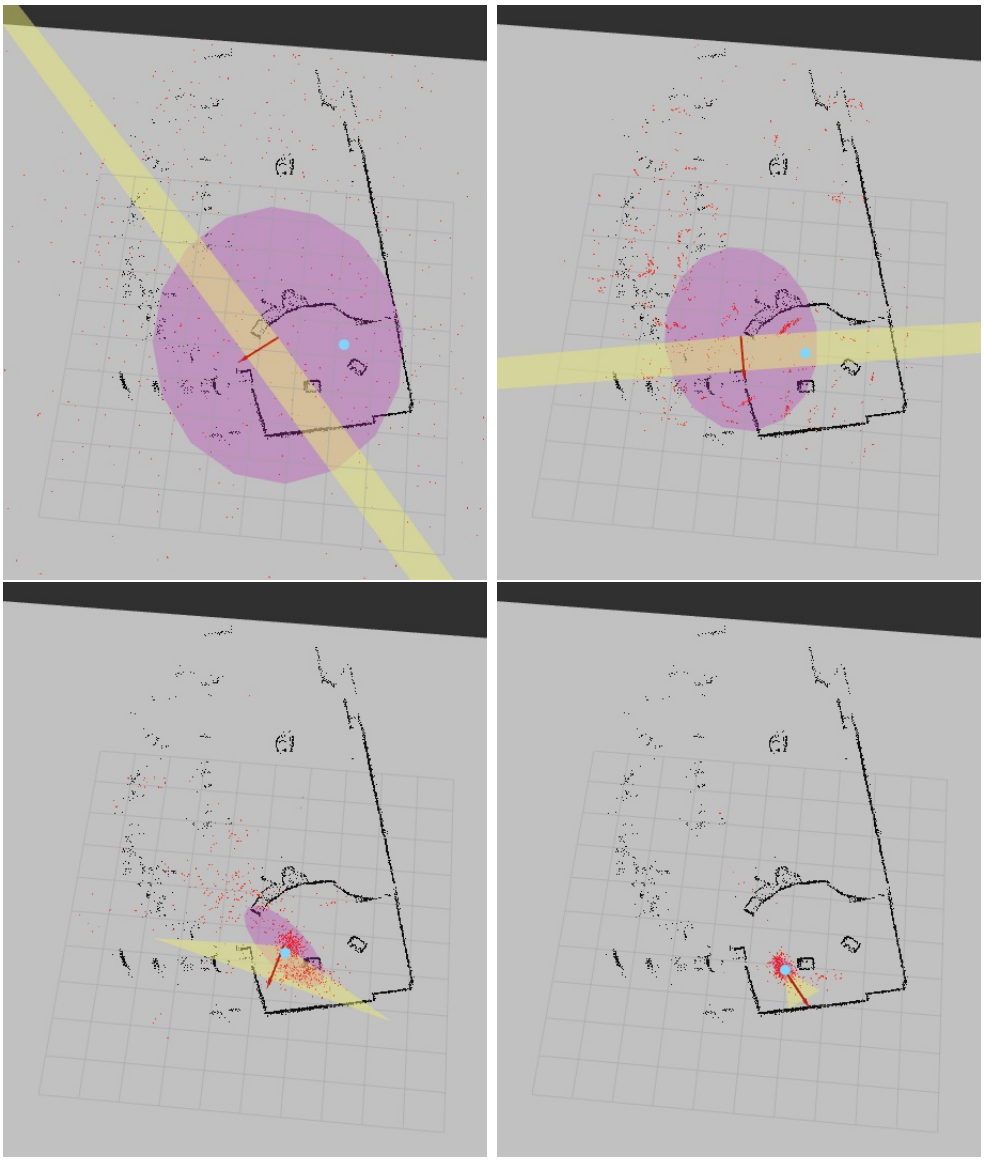


Figure 3: Convolution of Localisation (The blue dot is the actual robot position)

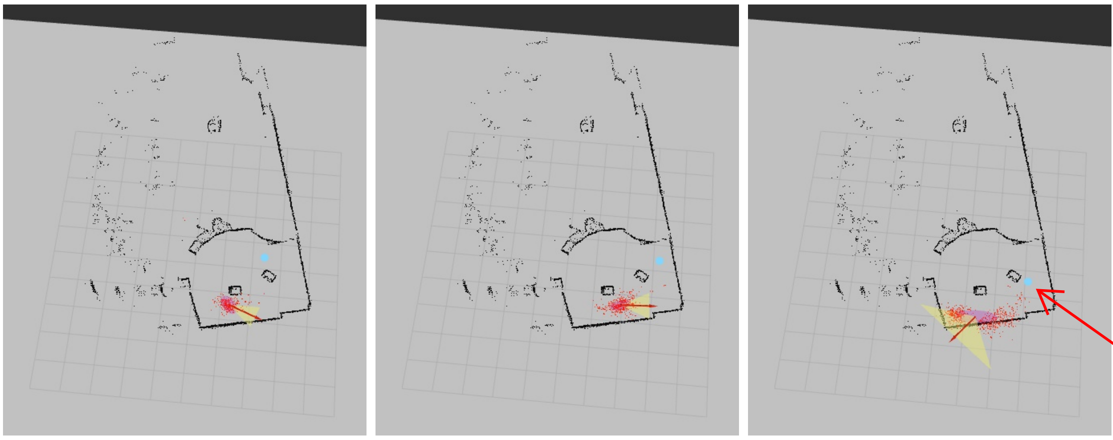
The uncertainty ellipse describes the area of uncertainty in which the robot could be located. The yellow line/triangle represents the orientation uncertainty. This means the robot could be positioned and rotated in the area of uncertainty and does necessarily matches the red arrow.

At the start, the particles are randomly distributed over the entire map. The estimated position spans nearly the entire map. And the estimated orientation is wrong. But as soon as the robot starts moving and the algorithm has multiple data points the particles, pose estimations and rotation start to convolute to quite an accurate pose estimation. After the robot drove about two meters the pose estimation was nearly precisely corresponding to the actual position of the robot.

15%

2. **You will now kidnap the robot! After the AMCL algorithm converges to a final pose, move the robot to another part of the map. Navigate around and see whether the AMCL algorithm can recover, namely to show the correct robot pose again. When the localization fails, try out the following three methods to recover from kidnapping and briefly explain the advantages and disadvantages of each method. (15%)**

   (a) Kidnapping without any help.



Very good!

Figure 4: Convolution after kidnapping (The blue dot is the actual robot position)

Without any help, the AMCL algorithm is not able to recover to a correct pose estimation. The results seem to get worse and worse.

(b) Setting the recovery alpha parameters.

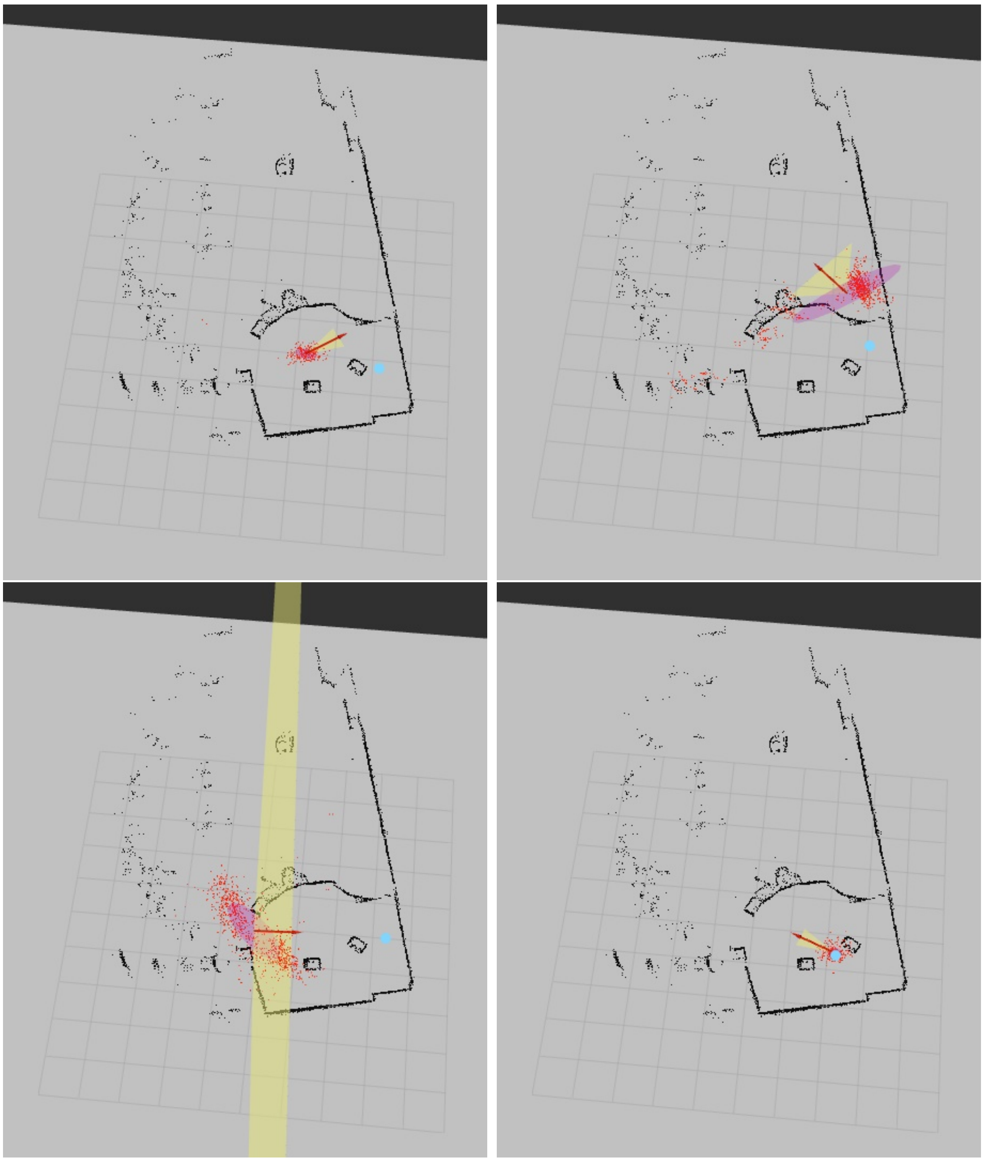


Figure 5: Convolution after Kidnapping with recovery alpha (The blue dot is the actual robot position)

With the recovery alpha parameters, the AMCL algorithm can recover the correct pose, but it took a very long, were to the area of uncertainty was very big.

(c) Pose estimate.

With the pose estimate, the AMCL algorithm was able to recover the correct pose fairly quickly. Also, the area of uncertainty was never really big. Therefore pose estimation is the best way to recover after kidnapping the robot, but the great downside is that one has to know the broad position of the robot in the world to use this solution.

(d) Reinitialize the global localization and redistribute all particles randomly on the map

This approach always works, because the AMCL algorithm start from scratch. One big downside of this solution is that the position needs to be newly calculated.

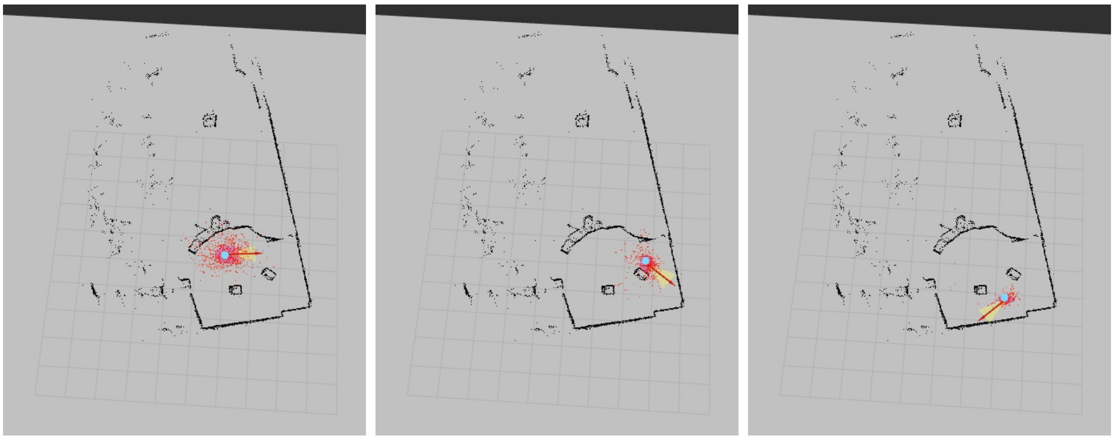


Figure 6: Convolution after Kidnapping with pose estimation (The blue dot is the actual robot position)

10% 3 Quiz: Battery Handling 10%

2% 1. **Batteries lose capacity over time. What do you need to do to ensure a long lifetime of the battery? List two measures you can take. (2%)**

   - Don't let the battery get empty or fully charged. Most batteries should always be above 20% charge. For storage, the batteries should be charged to vaguely 70% preferably using a charger that balances all the cells so they discharge equally during use.

   - Minimize exposure to extreme temperatures. Very high or low temperatures can lead to faster degradation of the battery components which can increase the safety risks like fire or explosion. This also means the battery shouldn't sit in direct sunlight. You should especially avoid charging in extremely cool temperatures.

2% 2. **You want to build a new, powerful wheeled robot from scratch! Name two safety precautions you would consider: (i) when it comes to selecting an appropriate placement for the batteries on the robot, and (ii) while mounting the battery onto the robot. (2%)**

   First of all, I would place the batteries low and near the centre (for equal weight distribution) with a container to keep them protected from potential physical harm and from sliding out of the robot, but still in an accessible position for easy charging. When mounting the battery onto the robot, I would keep the wheels in the air to avoid the possibility of the robot running off after connecting.

2% 3. **List two types of fire extinguishers. Which one can be used in the case of a fire caused by LiPo batteries? (2%)**

   - CO2 fire extinguisher: This is not effective in cleaning out metal fires, the fire could reignite.

   - Typ/Class D fire extinguisher: Can be used in case of a fire caused by LiPo batteries, because the used powder agent is effective at separating the ignited flammable metal from the oxygen.

2% 4. **Name two differences between LiPo and LiFe batteries. (2%)**

   - Lithium iron phosphate (LiFe) batteries are safer and recommended for self-build robots because LiFe batteries possess different chemical compounds which cause the battery to not burn as fast and ensure more robustness for physical stress. For instance, when a robot drives on rough terrain and receives some collision, the battery won't explode as quickly in comparison to LiPo batteries. Additionally, they are more robust against low voltages as well so they will also survive longer if they haven't been charged in a longer time.

- LiFe has lower voltage output and lower capacity than Lithium-polymer ion (LiPo) batteries.

2%

5. **On which ROS 2 topic can you check the status of the battery of your TurtleBot 3? What is the type of message used to report the battery status on this topic? (2%)**

   If you create/have a battery state publisher node, you can publish a sensor_msgs/BatteryState message (Battery State message) to the ROS 2 topic named /battery_status to create a battery state publisher. And you can check the output on the /battery_status topic with:

   ```
   ros2 topic echo /battery_status
   ```

4 Feedback

- How much time did you spend on doing this sheet per person? Anonymize your answer!

  Person 1: Questions: 30 min Practical: 60 min

  Person 2: Everything considered about 5 hours

- Was it too easy, easy, ok, hard, too hard?

  Person 1: Questions: easy Practical: ok

  Person 2: It was ok

- Please tell us what you liked in this exercise sheet.

  It was really interesting seeing the point cloud evolve.

- Did you face any difficulties? What should be improved?

  Setting up rviz2 for localization was a bit annoying because you had to start the localization server to configure it and then restart the server. After all, the map is only sent once in the beginning and doesn't get recognized if rviz2 isn't configured.

- ROS 2 Humble installation:

  - In the first tutorial, did you face any issues during the installation of ROS 2 Humble on your notebook?

    Person 1: No problems related to the installation as recommended only operating system-specific problems.

    Person 2: Installation went ok after I used the provided virtual box image.zip file, but I have problems with nodes in ROS 2.