CSCI-632 Assignment 4 Non-ROS

Pranshu Bheda

March 2020

Geometry

I assumed that for both the worlds the max distance in any direction is 15m. I used this value to normalize the x and y coordinates to a value between 0 and 1, so my resolution is 1/15. My map size is 400x400 with center at (200,200) giving me 200 units in each direction. I scale from world to pixel coordinates by multiplying the normalized values with 200 giving me the pixel coordinates.

Sonar

On using all of the sonar data the resultant map was very noisy owing to the reflections. To reduce noise I only decide to use the range of angles between -50 and 50 degrees. The inverse sensor model for sonar is proportional to the distance and angle hence on including that in my calculations the map became very clear.

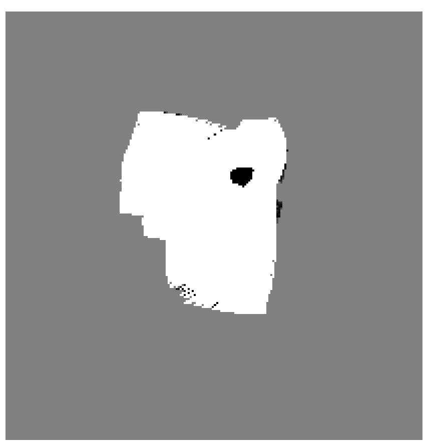

Figure 1: Small world sonar map

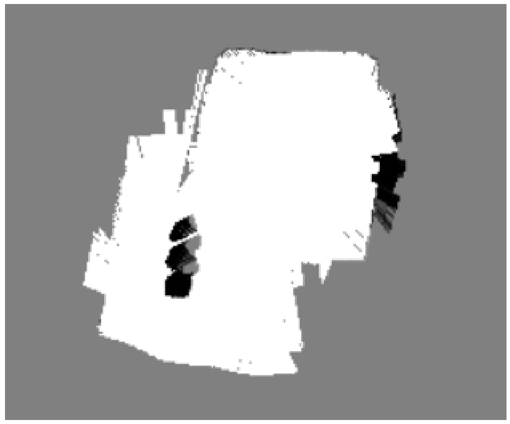

Figure 2: Big world sonar map

Laser

The sensor model for laser is very straightforward. It is very accurate hence it is not dependent on the angle and range. My sonar map is better than laser. I tried a few things to fix it like choosing the minimum sensed range in a range of angles i.e skipping some readings, reducing the count of log probabilities in case it gets too big, adjusting probability values but none seemed to work. It was understanding free space well, but not the obstacle boundaries as seen in the maps. My preliminary laser map looked like Fig. 3. After making some changes it looked like Fig. 4 where it misrepresented the obstacle as free space.

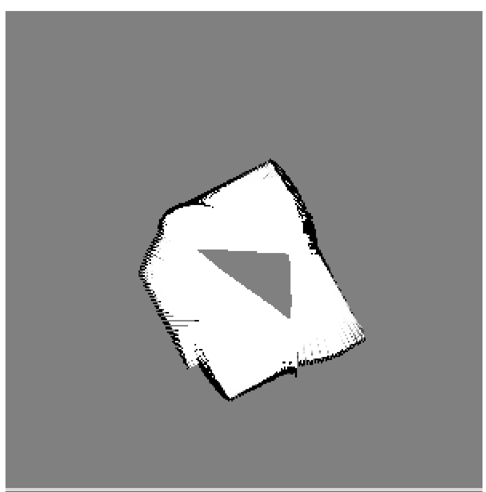

Figure 3: Initial Small world sonar map

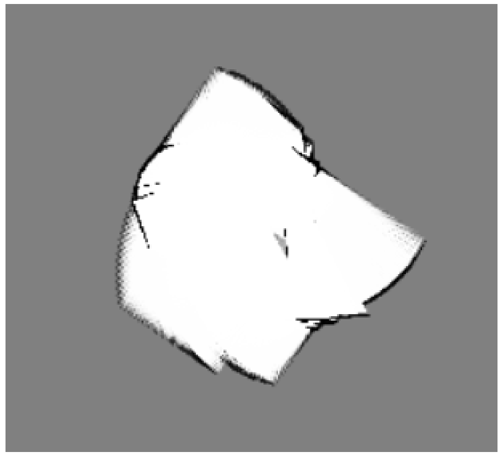

Figure 4: Small world laser map

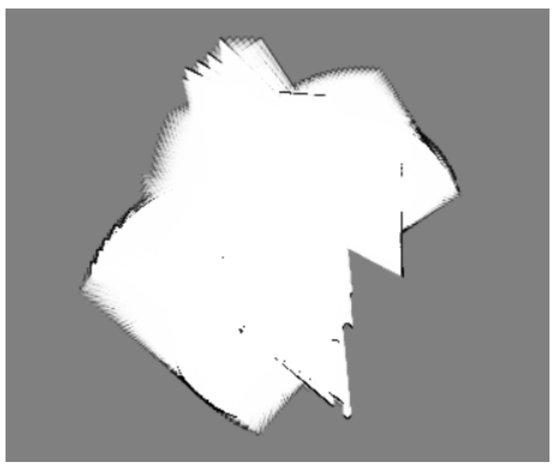

Figure 5: Big world laser map

Both sensors

For combining both the sensors we should ideally use the sonar readings to identify obstacles at close range and laser to identify free space since it has more range. In our implementation we consider these two readings to be independent and simply add their log probabilities up which gives us the following results. The results look a lot like using just laser.

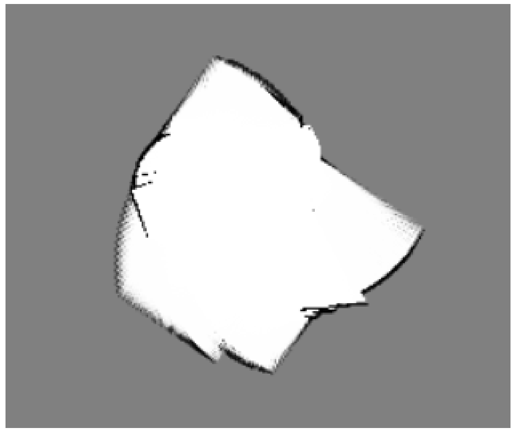

Figure 6: Small world both sensor map

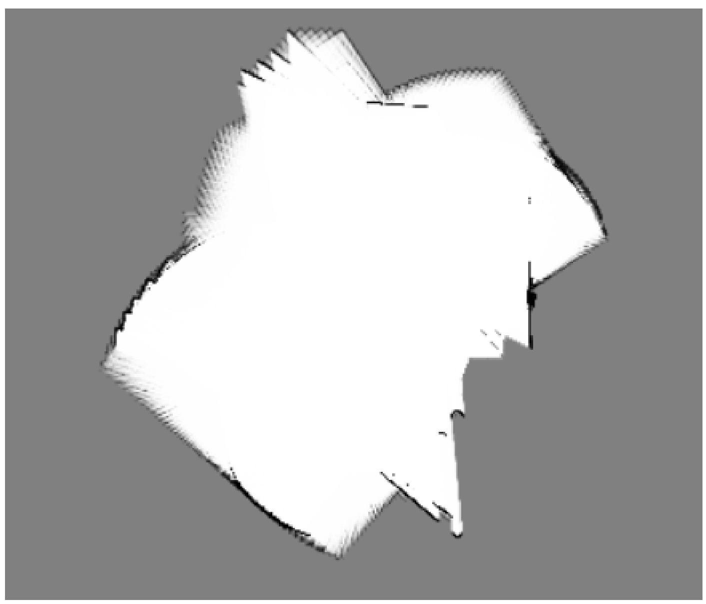

Figure 7: Big world both sensor map

Timings

The current operations of finding pixels that need to be updated can be improved by picking wedge of possible pixel coordinates using the pose and then check which of these lie in the sensor wedge. Our transformation operation to move from the simulation to the grid and performing update operations can be significantly improved using available matrix multiplication libraries.

Program

The image is not updated constantly but just generated at the end. I was testing this on the cs machine remotely and showing the image after each update made it very slow.

```
Example:-
python mapper.py sonar big-world-data.csv
Params:-
<sensor> - {laser, sonar, both}
<file> - {big-world-data.csv, small-data.csv, data.csv}
```