IN4254 Assignment 3

Hang Ji
Delft University of Technology
Delft, The Netherlands
student number: 4759745

Zhao Yin
Delft University of Technology
Delft, The Netherlands
student number: 4778863

I. Basic Information

The android application is implemented on `HUAWEI ANE-AL00 (Android 8.0.0, API 26)`. The libraries used in this project are `apache.commons.math3` and `PolyUtil`.

II. Bayes Localization

The Bayesian method is the same as in the second report. The layout is shown in Figure 1. The confusion matrix of online evaluation on the smart phone is shown in Figure 2. The training data for the WiFi RSS fingerprint was collected at different time during three days. The online evaluation was carried out at different time during three days.

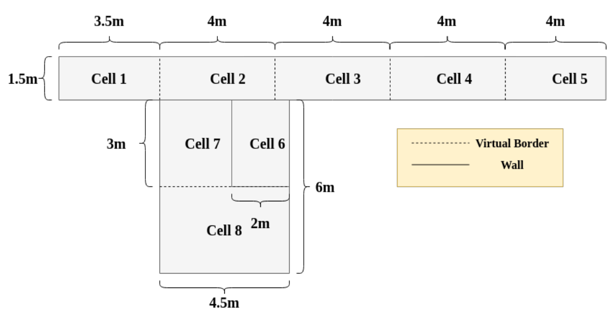

Fig. 1: Layout

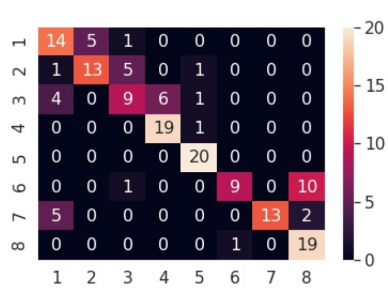

Fig. 2: Online evaluation confusion matrix

III. Particle Filter Localization

A. Motion Model

The direction of the motion model is measured by `TYPE_ROTATION_VECTOR`. The number of steps is measured by `TYPE_ACCELEROMETER`. The calculation of the coordinate is shown in the Equation 1. `Offset` is the angle between the north pole and the x axis of the layout. `Theta` is the angle between the north pole and the moving direction. `AB` is the moving distance. `Offset` and `Theta` are measured by `TYPE_ROTATION_VECTOR`. The implementation of step counter will be explained in Section IV.

$$\begin{aligned} x_2 &= x_1 + ABcos(theta - offset) \\ y_2 &= y_1 + ABsin(theta - offset) \end{aligned} \tag{1}$$

$$AB = num\_step \times step\_size \tag{2}$$

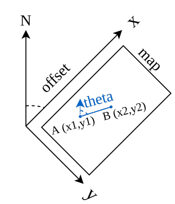

Fig. 3: map

B. Implementation of Map

The implementation of the map is shown in Figure 4. 100 pixels represent 1 meter on the map. Black lines represent real walls and green lines represent virtual walls.

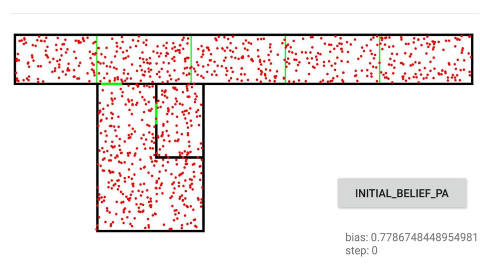

Fig. 4: Implementation of the map

C. Implementation of Particle Filter

We implemented a polygon to constrain the positions of the particles. Particles are first randomly sampled in the whole screen then only ones within the polygon will be kept (using Google map utils, i.e., `PolyUtil`). Particles maintain a table to store the current and previous positions in order to get its moving trajectory.

Aware of the moving trajectory, we can detect whether the particle goes through a wall by judging whether the trajectory line has intersected with the wall segment. We detect the violating particles after every step to ensure the precision and resample those invalid particles on the valid ones. We see the variance of the particles locations lower than a predefined threshold (in our case, e.g., 4000) as converged. After convergence, our implementation will predict our location after every step by calculating the centroid of all particles. If all particles become invalid, they will move to the left top position without further changes, and so as the centroid.

D. Results

The blue line in Figure 5 is the unique path to reach convergence of the tracking point. After reaching convergence, the tracking point is able to follow the movement of the user and present the current location on the map. One example of the tracking point is shown in Figure 6.

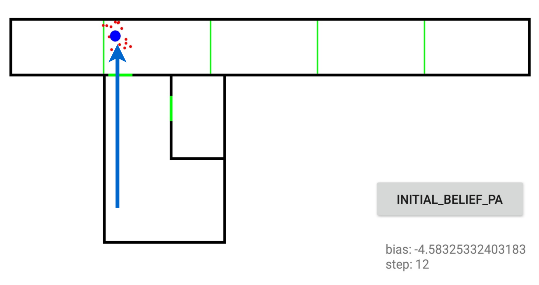

Fig. 5: Cell 2

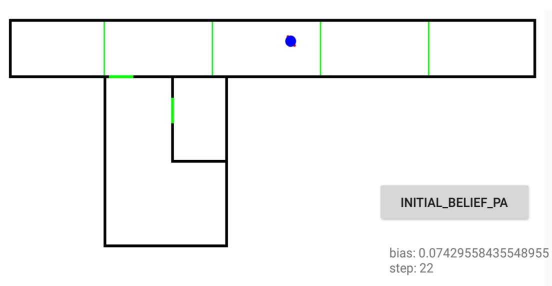

Fig. 6: Cell 3

IV. Novelty for Particle Filters

The step counter is implemented by the accelerometer. The acceleration of y axis while walking is presented in Figure 7. The step can be detected by the number of positive threshold crossing on the waveform of ay. The positive threshold crossing can be detected when the current ay is larger than the threshold and the previous ay is smaller than the threshold. The threshold chosen for detecting the peak is 0.18.

The step counter implemented by the accelerometer has the following advantage: It is able to detect the step with low computation latency compared with the step counter implemented by `STEP_COUNTER`.

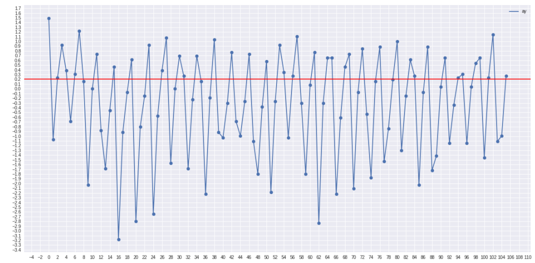

Fig. 7: Acceleration of y axis while walking

V. Individual Workload

The individual workload is presented in Table I.

	Hang Ji	Zhao Yin
PF Motion Model and Map	×	
Implementation of PF		×
Bayes AP filtering	×	
Bayes Calculation		×

TABLE I: Individual workload

VI. Challenges & Possible Future Directions

The challenges and possible future directions in this project are discussed in the following points:

- **The drift error of magnetic offset:** It is difficult to measure an accurate and steady angle between the north pole and the walking direction due to the drift error of magnetic offset.
  One possible future direction to eliminate the drift error is by applying **Kalman filter** with sensor fusion. Kalman filter is suitable for this application scenario since it can run in **real time** with the present input measurements and the previous calculated state and its uncertainty matrix.
- **The limitation of the step counter in certain walking patterns:** The step counter only takes the acceleration of ay into consideration. It can only record the accumulated steps correctly when the user holds the smart phone in a certain posture. The step counter by counting the number of threshold crossing is sensitive with abrupt changes in ay.
  We improve the algorithm of step counter by the following method: firstly, reduce random noise of ax, ay and az with the moving average filter; secondly, compute the square root of Equation 3; thirdly, obtain the final feature for the step counter by reducing random noise with the moving average filter; at last, one step can be detected by the positive threshold crossing on the waveform: $prev\_feature\_value < threshold$ and $current\_feature\_value > threshold$. The value of threshold is 5.3

$$filtered\_ax^2 + filtered\_ay^2 + 0.2 * filtered\_az^2 \quad (3)$$

The waveform of accelerations before the moving average filter and after the moving average filter are shown in Figure 8. The waveform of the feature before the moving average filter and after the moving average filter are shown in Figure 9.

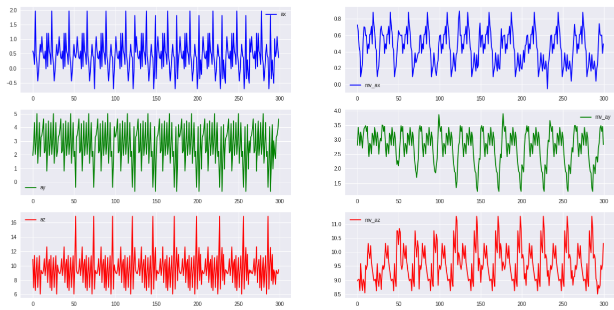

Fig. 8: Accelerations

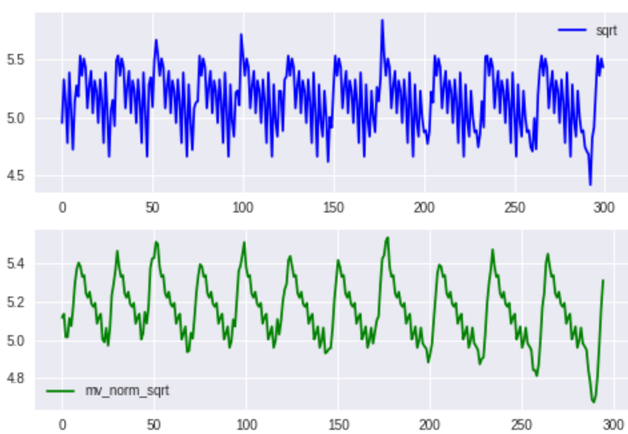

Fig. 9: Feature

The moving average filter is the most common filter in DSP. It can reducing random noise while retaining a sharp step response. The improved method is able to taken consideration of accelerations in different directions. It also retains the characteristic of low time latency in counting the number of steps.