An Indoor Navigation and Localization System

Qi-wei Bao, Chris Papachristou, Frank Wolff
Case Western Reserve University
Cleveland, Ohio 44106, USA
{qxb16, cap2, fxw12}@case.edu

***Abstract*—This paper introduces a mobile navigation and localization system suitable to guide visitors in environments indoors without GPS help. This guidance is especially useful in large multilevel structures such as hospitals, shopping malls, airports, and can be beneficial for people with special needs. The system is implemented on smart phone platforms equipped with several embedded sensors. The proposed localization technique is based on a combination of (a) Wi-Fi signal strength capturing from pre-built sample area points and (b) a pedestrian step and stride identification using embedded sensory data. The method has been implemented and tested in a large hospital building.**

I. Introduction and Motivation

There is increasing demand for real-time tracking and routing of individuals within multistory structures accessible to the public such as hospitals, office buildings, underground parking garages and airports. One challenge for indoor positioning and navigation technology is that there are some structures inside buildings that need to be considered such as revolving doors, wheelchair ramps, sloping ceiling, elevators and stairs, and other such factors.

Several technologies have been proposed for indoor navigation and localization services such as Wi-Fi based positioning and PDR (Pedestrian Dead Reckoning). However, devices based on Wi-Fi fingerprint positioning usually have poor stability due to the impact of complex surroundings inside buildings, and PDR produces accumulated errors.

One example scenario concerns how to help a visitor in a large hospital (e.g. Cleveland Clinic) finding a route to a place such as an office, a patient room or a waiting room. The intent of the navigation system is to provide directions to reach the destination but the route may have to go through long and often winding corridors, up and down elevators, or even cross building bridges. These directions need to be real time considering all last minute changes in the surroundings. Getting direction information by calling large organizations is not so efficient today.

In this work, a navigation and localization system is proposed suitable for large building structures such as large hospitals. A hybrid self-adaptive technique was employed for the user's localization and an optimized path planning discovery algorithm for navigation. The system first estimates the user's position by calculating the direction and stride of each step. A grid-oriented Wi-Fi RSSI (Received Signal Strength Indicator) method is employed while, in the meanwhile, collecting the user's location information. After that, the K-Nearest Neighbor (KNN) algorithm is employed on the RSSI data to provide a feedback correction process.

The system navigation software can be embedded in a small mobile device with simple interface making it useful for visitors to find their way in large buildings. The mobile device should be equipped with acceleration and gravity sensors, similar to the ones used in current smart phones.

II. Background

Indoor positioning and navigation have become a very popular application area in recent years. Several technologies have been employed to support this application. The most important existing technologies potentially suitable for indoor positioning are [1], [2]:

1) Using signals of radio communication technologies, either those already widely deployed on mobile devices such as wireless local area networks (WLAN) and Bluetooth, or those generated by new and dedicated infrastructure (RFID, ZigBee).
2) Self-contained sensors embedded in mobile devices including but not limited to: accelerometer, gyroscope and magnetometer.
3) Magnetic field fingerprinting using an indoor map (building floor plan).

However, none of these technologies alone can provide a complete and reliable indoor positioning solution on a single mobile device independently. For example, the use of RFID signal based solutions requires additional RFID readers and tags. These components are currently not embedded in mobile devices like smartphones [1].

Several researchers have compared the performance of indoor positioning systems under different technologies. Localization systems can be classified into active and passive systems [3]. Active systems require the tracked person to carry an electronic device for information sending or receiving to participate in the tracking process [3], [4].

While in passive localization, the position is estimated by measuring device signal or through video process. Most techniques for passive localization require complex differential measurements, which is limited by the hardware of the mobile devices. Therefore, active localization systems, due to this

978-1-7281-1416-3/19/$31.00 ©2019 IEEE

problem, are more widely used than passive localization systems. Recently, two technologies, wireless-based positioning and PDR (Pedestrian Dead Reckoning) technologies have been applied as on active indoor positioning systems.

Regarding wireless-based techniques, there are two design issues. First, build a wireless location system based on network infrastructure and Wi-Fi routers, b) use existing wireless network infrastructure to locate a target. Although the first technique offers maximal flexibility, it may not be feasible in large buildings that already contain their own Wi-Fi infrastructures. Either way, radio communication technologies such as include Wi-Fi, Bluetooth, ZigBee can be used. However, there are disadvantages of indoor localization methods based on RF signals as they are subjected to to multipath, positioning environment, human body and other interference. A key metric for distance estimation in wireless technologies is based on the Received Signal Strength Indicator (RSSI) [5], [6] $RSSI = -10\,n\log_{10}(d) + A$ where d is the distance from signal sender to signal receiver, A is the received signal strength at one meter of distance, n is path loss exponent.

In reality, the user's location is not described in one dimensional space. Thus, collecting radio frequency signals to/from the wireless router is not enough for the RSSI approach. However, RSSI can be improved by triangulation and fingerprint methods [7], [8].

Another method proposed recently employs inertial and other self-contained sensors for user positioning [9]. The so called Pedestrian Dead Reckoning (PDR) algorithm is applied which involves the following: step counting and step length and direction estimation, motion classification, transit mode detection, and floor change detection in multi-level buildings. The user's position can be estimated through these tasks. By using this algorithm, the positioning and navigation system can retain a lower deployment costs for infrastructures [10].

A disadvantage of PDR is that its accuracy highly relies on the sensors. Typically, smartphone sensors do not have very high precision thus PDR alone may not be suitable for indoor positioning and navigation due to the accumulation of position, velocity and attitude errors. Another challenge for PDR is the existing angle between the heading direction between the mobile device and the direction the user is moving. Some mitigation techniques to address this issue have been proposed [11].

III. Approach

In our work, a hybrid approach was proposed which not only integrates the Wi-Fi RSSI fingerprint with the PDR techniques, but it builds an automatic self-adaptive system using the received Wi-Fi signals to correct the user's stride estimation based on the PDR techniques.

Our proposed indoor navigation technique involves several parts shown in Fig. 1. System pre-processing (includes base station and sampling points selection, and location and sensor databases building), Fig. 2. Collect the visitor's Wi-Fi signal strength data through a grid-oriented method based on the Wi-Fi Received Signal Strength Indicator (RSSI), Fig. 3. Estimate the user's position through the PDR method which entails detecting the user's step and calculating the direction and stride of each step, Fig. 4. Apply the K-Nearest Neighbor (KNN) algorithm on the detected Wi-Fi signal strength data, and the Wi-Fi signal strength data near the location estimated by the PDR method, Fig. 5.

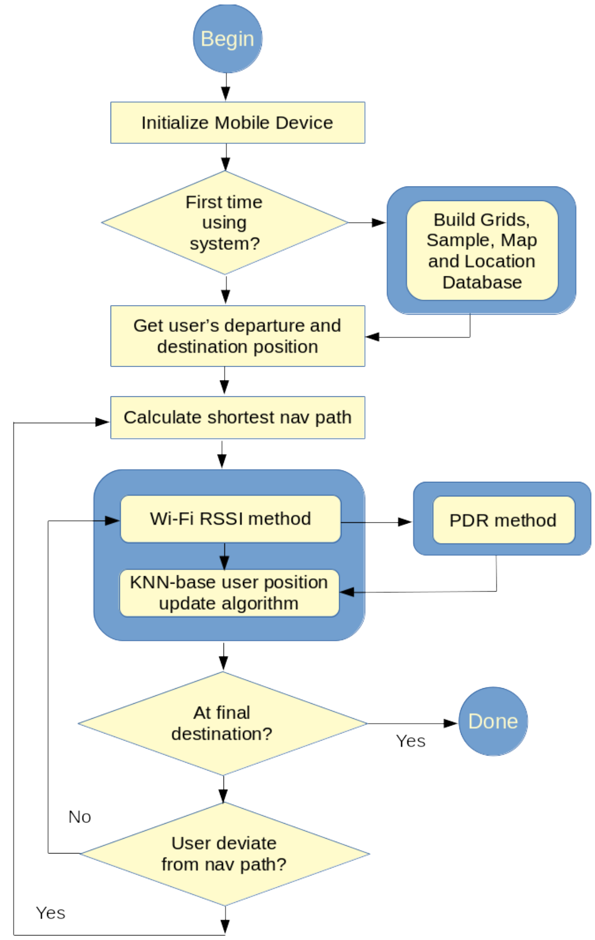

Fig. 1. Approach

A. Corridor grids and Wi-Fi sampling

During system pre-processing, a location database is designed based on the blueprints of the building floors. It is assumed that Wi-Fi router devices are already installed in each floor. For each floor, some routers were chosen to work as base stations. Since the user's are walking on the corridors during navigation, the corridors are divided into a list of grids with grid labels. The grids include special nodes such as corners, cross-roads, special doors, such as stairs and elevators. Ordinary grids are corridors divided by built-in area. Fig. 6 shows how to divide corridors to grids. And the grids setup follow the corridors. If the corridor is not straight, the grids

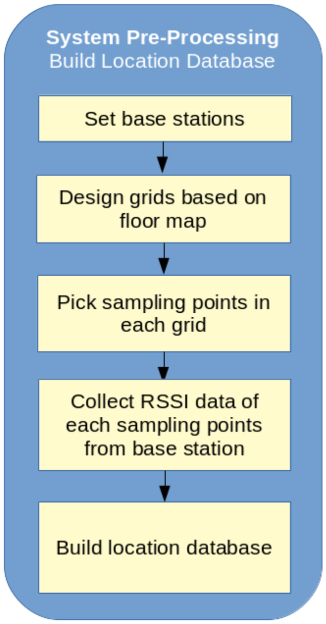

Fig. 2. System Pre-processing

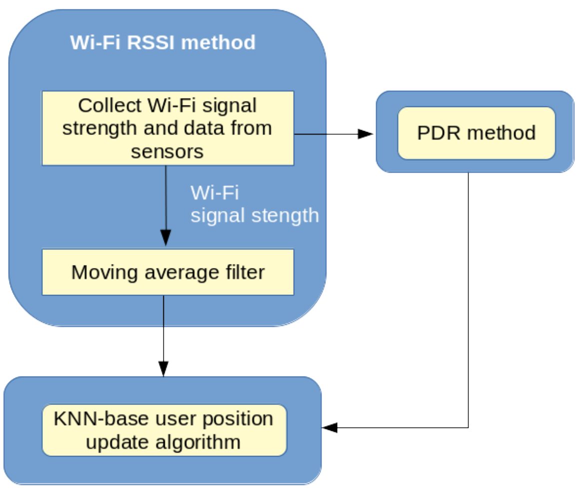

Fig. 3. Wi-Fi RSSI method

are set following the shape of corridors, but with the same spacing.

In each grid, some local points are picked for sampling to provide location information and their corresponding Wi-Fi signal strength from all the base stations. These data, together with the grid labels for each sampling point, are used to built a location database.

Another emphasis in the preparation work is to collect sampling points on the corridors as shown in Fig. 7. The Wi-Fi signal strength data from the base stations to the sampling points are collected in advance as fingerprints for later comparison. Obviously, the density of sampling points

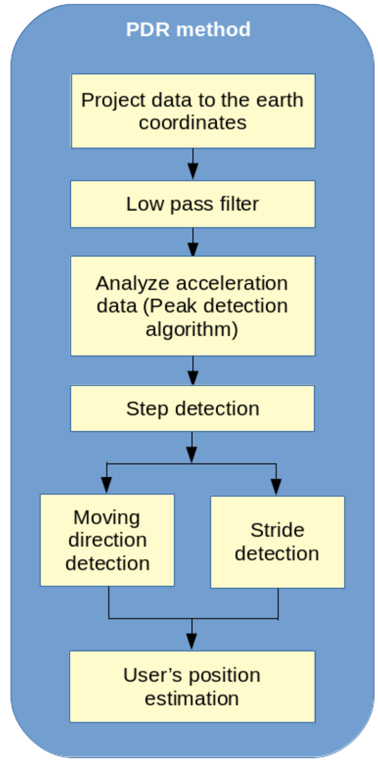

Fig. 4. PDR method

(how many sampling points to be picked in certain area) directly influences the accuracy of user positioning. But too many sampling points may require a lot storage. To make sure the layout of sampling points fully covers all the corridors, first, grids were set on the corridors, and then sampling points were selected in each grid. In the middle of each grid, there is one sampling point, so that the sampling points can cover all the corridors.

We illustrate how to represent the location of sampling points in Fig. 7. There are two corridors shown. Corridor C is straight while Corridor A is not. Sampling points α and β are located in Corridors A and B, respectively. The yellow cycles represent the Start points of each corridor. And the orange triangles are the location of sampling points α and β whereas x_β and y_β denote distances from the Sampling point β to the Start point of Corridor D. If the coordinates of Start point of Corridor C are (*XC*, *YC*), then coordinates of β are $(XC - x_\beta,\ YC + y_\beta)$. More generally, if θ is the angle between the y axis of the corridor and the direction of Sampling point α to Start point of Corridor A, then the sampling point coordinates of α are:

$$XA - y_\alpha \times YA \times sin\theta\,;\ YA + y_\alpha \times cos\theta$$

When the user starts operating the system, after initializing the hardware on the mobile device, the system would check the user's current departure localization. Then, as the mobile

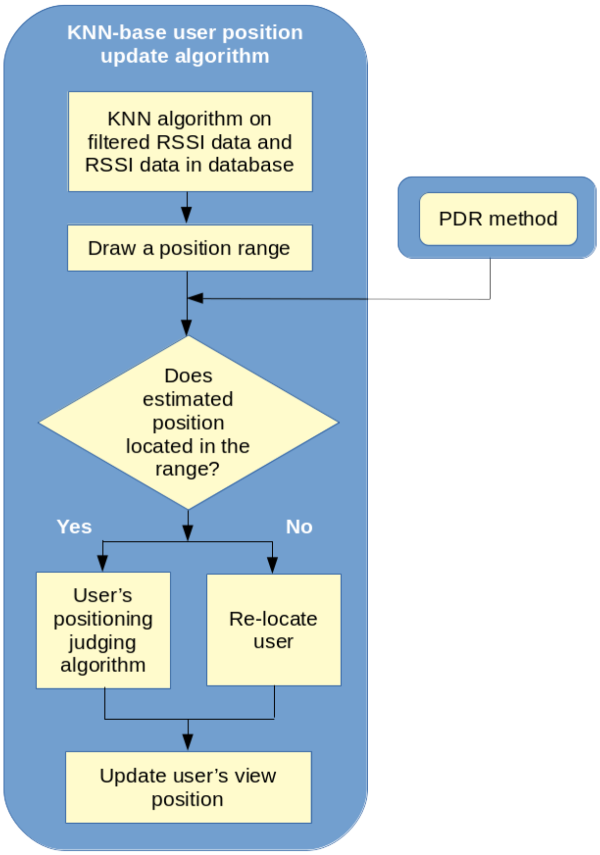

Fig. 5. KNN-based user position update algorithm

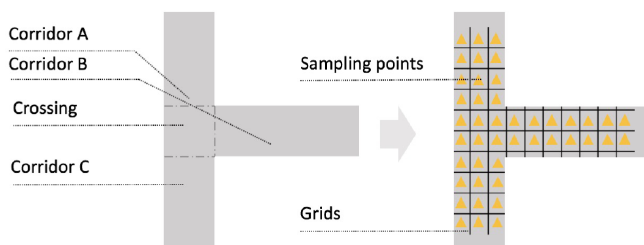

Fig. 6. Floor grids

device user is holding the device, the system will collect automatically the following type of data: (a) the Wi-Fi signal strength from nearby base stations, and (b) data about the user's movement which include acceleration, gravity data, and magnetic data, based on the coordinates of the mobile device. The Wi-Fi fingerprint positioning is applied on Wi-Fi signal strength data from the base stations for a rough estimate of the area the user may be located, Fig. 1.

B. The PDR method

The PDR method is illustrated in Fig. 8. Once the user's departure position is known, the position after the first step can be estimated using the stride, i.e. the length of one step, and the direction of one step. Suppose the departure position

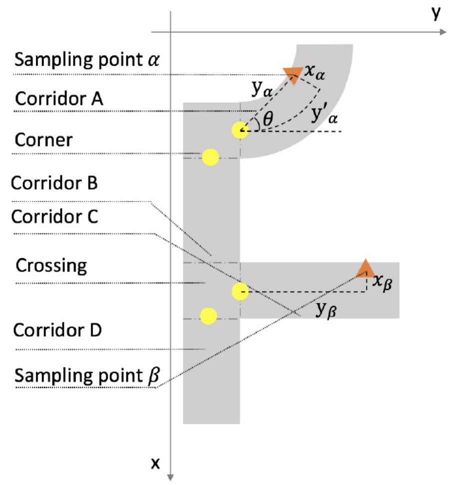

Fig. 7. Selecting sampling grid points

is $(Location_{x0}, Location_{y0})$. Then the stride of the first step *Stride1*, and the angle between the moving direction and positive direction of x axis in the location coordinate is θ_1, the user's position after the first step is described by

$$\begin{aligned} Location_{x1} &= Location_{x0} + Stride1 \times cos\theta_1 \\ Location_{y1} &= Location_{y0} + Stride1 \times sin\theta_1 \end{aligned}$$

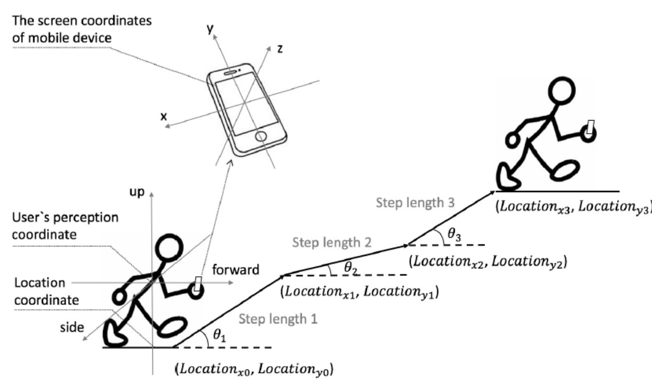

Fig. 8. Application of the PDR technique

Shown in Fig. 9 are the main stages in the PDR method. First, acceleration, gravity, magnetic data and angular velocity are collected from the sensors of the mobile device apart. These sensory data are used to detect the user's step, direction and stride. To detect the stride, the system first detects if the user makes a step. Then, for each step, the stride can be estimated using a stride detection algorithm [12]. The algorithm uses motion acceleration data to calculate the length of the steps. In the meanwhile, after projecting data from the magnetic sensor and the gyroscope to the earth coordinate system, the user's moving direction can be estimated.

There are a number of issues that needed to be resolved:

– There are differences among the screen coordinates of mobile devices, the user's perception coordinate, and lo-

cation coordinate saved in the system, Fig. 8. The system needs to transform the data among different coordinates.
- A step detection algorithm needed to detect if the user takes a step.
- Also, the moving direction and length of each step need to be estimated for better description of each step.

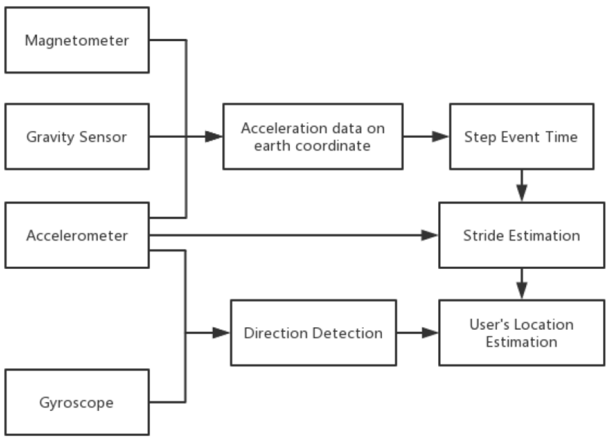

Fig. 9. The PDR sensor system

The data collected from all these Android sensors are based on the mobile screen coordinates. When the orientation of the device screen is changing, the mobile coordinates will change along with the device. The default orientation in this coordinate system is defined relative to the screen of the phone. The axes will not swap while the device's screen orientation changes. Specifically, the X axis is horizontal from the screen and points to the right of the screen; the Y axis is vertical from the screen and points to the up of the screen; the Z axis points towards outside of the screen [13].

To map the device (screen) coordinates to the earth coordinate system, a linear vector projection technique is used, as shown in Fig. 10, the details of which are in reference [14].

C. Step detection based on the human gait

Due to the characteristics of human walking, which is cyclical and repeatable, the system identifies if the user advances a step by monitoring if there exists a wave crest in the acceleration data.

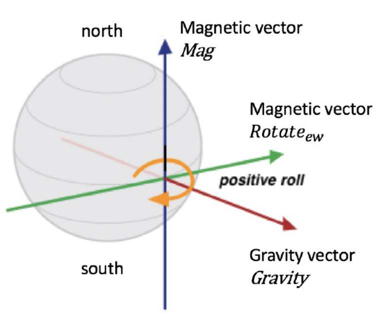

Fig. 10. Mobile coordinate mapping to earth coordinates

Fig. 11 depicts the gait cycle from the vertical view, corresponding to how the velocity and acceleration would change while a user is moving. Assume in the beginning that both feet are on the floor, i.e. the velocity is zero. When the user starts walking, one foot first moves forward, while the user's velocity increases and the acceleration is above zero. Then, when this foot is moving down to the ground, the velocity is also reduced. The other foot will move only after the first foot reaches the floor (just walking, no jumping). This means there exists a time instant when both feet are on the floor and the velocity is zero. Thus, during one step, the user's velocity is first increasing, and then decreasing.

Also, during a user's walking, there exists a vertical oscillation during each step, as marked in Fig. 11. And when the user reaches the highest point in the vertical oscillation, the velocity is also the highest, while the vertical acceleration is equal to or larger than zero. Note the acceleration vector vertical size, $AccEarth_z > 0$, when the user is accelerating, otherwise $AccEarth_z < 0$.

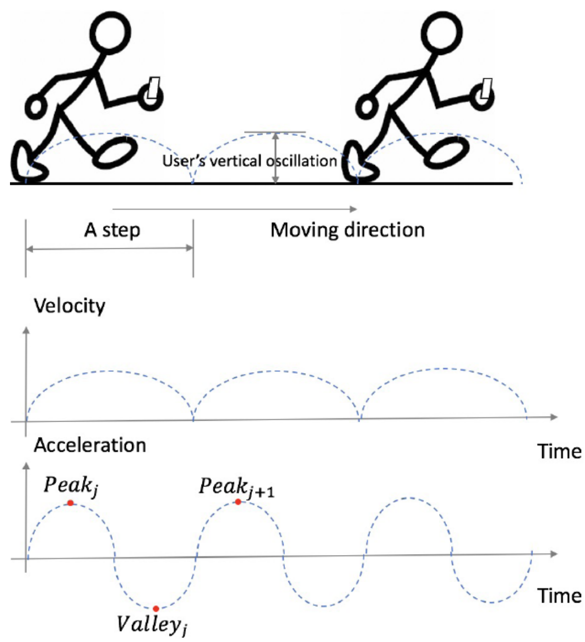

Fig. 11. Pedestrian movement detection

An algorithm is proposed to calculate the direction of walking by integrating both the gyroscope data and acceleration data projected on the horizontal directions. Last but not least, the stride of each step is calculated using a stride detection algorithm. A second rough position the user may be located is estimated from the direction and length of the user's step (stride). Details of these calculations are in reference [14].

D. Estimating the user's position

To update and estimate the position of a user, we first need to filter the Wi-Fi RSSI data to exclude noisy data. Then we need to apply the K-nearest neighbors algorithm (KNN) to

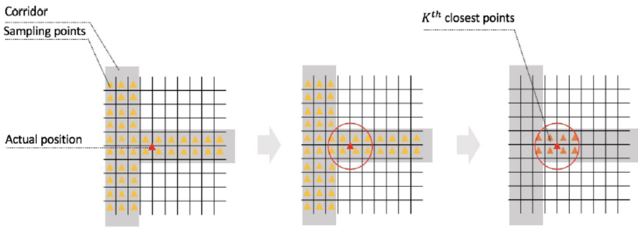

Fig. 12. Coordinates of K nearest points

determine if the area from the Wi-Fi fingerprint positioning method matches the location estimated from PDR method.

Clearly, the signal strength received from a Wi-Fi router is not simply linearly correlated to the physical distance between the router and the Wi-Fi module for detection Wi-Fi. And if a map is drawn to describe the signal strength of one Wi-Fi router, the change of signal strength value should be monotonically. This is because indoors the movement of a user progresses in consecutive steps. Thus, the received signal strength should not exhibit temporal fluctuation.

Suppose a user is walking straight, there is only one router and the user's mobile device is receiving signals from the router. If there exists a signal strength value too far from the its value at the last moment (the difference being bigger than a threshold), this input signal should be ignored. A moving average filter can be used to remove these noisy signals. Actually, there is a limit on the times of scans per time unit an access point broadcasts to a mobile device which is roughly every 100 ms. Thus the Wi-Fi module needs to maintain at least 100 ms interval between two scans. Since the mean time for an average human stride is about 675 ms [12], around 6 times Wi-Fi signal strength would be received from the access point during one step. The moving average filter needs to be applied on a set of received signal strength which is equal-time symmetrical from the detected time of one step, as follows,

$$\mathit{RSSI}(i) = \frac{1}{M}\sum_{-(M-1)/2}^{(M-1)/2} x(i \times M + j)$$

where M is the number of input signal points for averaging, i is the index of the output signal, j is iterated from 0 to M for input points counting, $RSSI$ is the base station output signal, and x is the input signal.

From the location database, sampling points near the location estimated from the PDR can be chosen. For each input case, select K cases from database that is closest from the input case. Fig. 12 shows how KNN works. Suppose the red triangle sign represents the user's actual position, the yellow triangle signs represent the positions of sampling points in the grids described earlier. For each of the sampling points, we can get its corresponding received $RSSI$ from different base stations, i.e. $BSdetect_1, BSdetect_2, ...$ and so on.

The KNN algorithm calculates the Euclidean distance from the Wi-Fi signal strength data. The latter is collected from the Wi-Fi module on the mobile device and the Wi-Fi signal strength data of the sampling points. After calculation, we sort the Euclidean distances and select K sampling points with shortest distances if the Euclidean distances of these points are smaller than a tolerable standard value, determined by experimentation.

If the distances are within the thresholds, the estimated positions are regarded as the user's current position. However, if the distances are too large, the system might need to collect data again.

E. Navigation path method

An optimized path finding algorithm for navigation is also proposed in this system. For illustration, Fig. 13 shows the map of the first floor of Cleveland Clinic Avon Tower. The aim of the system is to guide the visitors holding a mobile device from the arrival position to their destination. A navigation path is also provided in the figure. While the user is walking, the system keeps following the user and updates its new positioning in time.

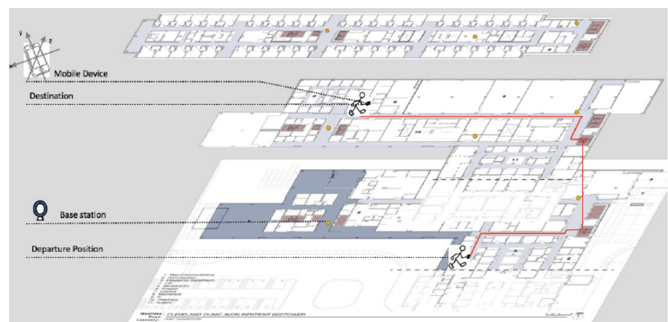

Fig. 13. Navigation path in Cleveland Clinic floor

To simplify the problem, the whole buildings are abstracted by graphs of vertices and edges. Vertices represent positions the user may approach such as doors of rooms, front desk, elevator doors, and stair doors, and doors of parking lots. Edges connect vertices including corridors, stairs, as well as bridges between buildings. The edges are weighted by their lengths. Fig. 14 shows how to represent the floor map by a graph.

A recursive algorithm is used here to find out an optimal path between two vertices, representing the navigation start and destination. The core of the algorithm is based on the well known weighted shortest path techniques, for example the Dijkstra algorithm with Fibonacci heap [15], [16]. When the user starts using the navigation system, an optimal path is determined using this system for guidance. Once the user is moving, the system continuously checks if the navigation path is followed. If for some reason the user is not on the planned path, the system would run this algorithm to reschedule a new optimal path based on the current position.

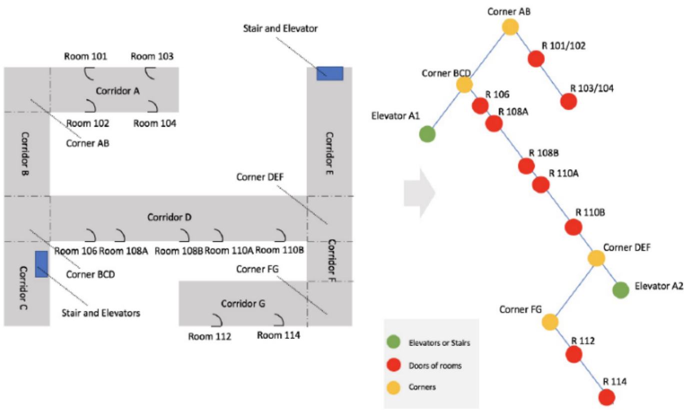

Fig. 14. Floor map graph representation

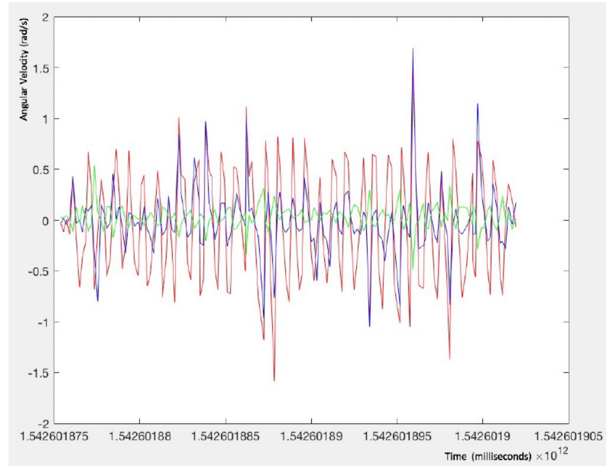

Fig. 16. Angular velocity data.

IV. Experiments and Results

In our experiment, there are two major hardware components: the base stations (Cisco Wi-Fi routers) and the mobile devices. The Samsung Galaxy Note II is an Android mobile device and is equipped with built-in accelerometers, gyroscopes and magnetic sensors, as well as a Wi-Fi module. The software we developed runs on Android is written in Java. The version of Android SDK used for development is 26.0.2.

Some results are depicted in Figs 15 and 16 where the x-axis shows system time in millisecionds. The acceleration unit is m/s^2 whereas the angular velocity unit is radians/sec.

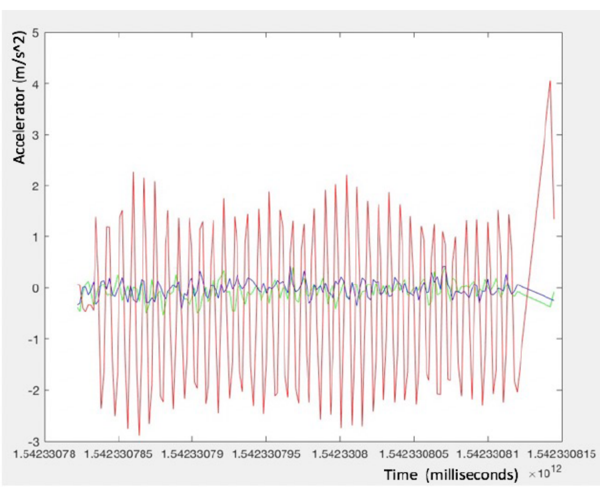

Fig. 15. Acceleration data.

To estimate the accuracy of step detection we tested many walking trips of at least 40 steps, while recording the step peaks. This is shown in Fig. 17 which marks all the peaks of steps detected from the experiment as a result of the peak detection algorithm mentioned earlier.

Our navigation system also detects direction of user's motion, particularly for straight walk and turning. In the direction estimation, we judge there is a turning only if both the acceleration data and the angular velocity data change sharply. From the experiments, we tested 20 times turning left and 20 times turning right. Fig. 18 shows results of turning left.

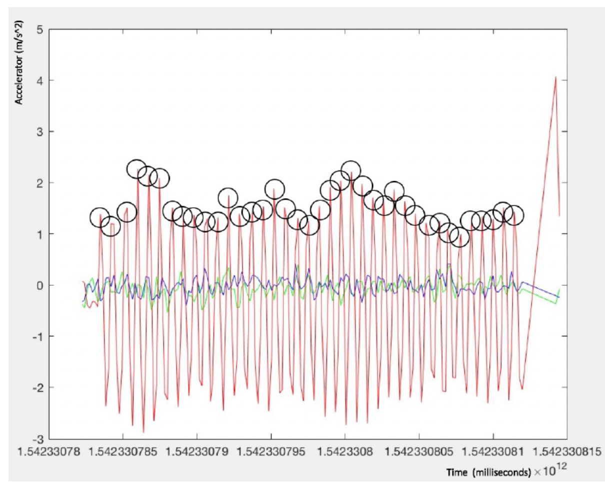

Fig. 17. Peak step detection

Finally, to estimate the accuracy of stride, we use a piece of rope to limit the length of each step, with the length of a stride limited to 0.5 meters.

Experimental results show that the proposed approach can achieve an overall location error of about 1.67 meters within 20 meters user's walking distance (accuracy: 92%). In comparison, using only the PDR method, the error is about 2.31 meters (accuracy: 88%).

V. Summary

In this work, we presented an indoor system for localization and navigation system based on Android mobile devices. A hybrid approach was proposed for localization using both the Wi-Fi RSSI fingerprint method and the PDR method. A shortest path navigation algorithm is also implemented after shortest-path optimization.

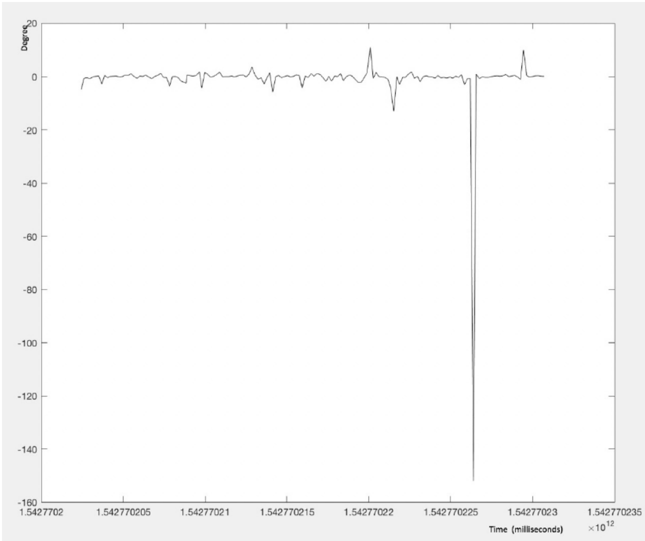

Fig. 18. Direction detection – turning left

The accuracy of Wi-Fi RSSI fingerprint may not be very high because the Wi-Fi signal strength could be influenced by the environment. The PDR method, on the other hand, returns a more accurate result but there exists an error accumulation problem.

We analyzed the reasons that may influence the accuracy of either the Wi-Fi RSSI fingerprint method or the PDR method and tried to eliminate these problems: For the Wi-Fi RSSI fingerprint, we added a moving average filter on the RSSI detected process to exclude the possible noise data.

For the PDR method, all the coordinates in the system were unified to make sure that all of the data to be processed are based on the earth coordinates. Filters were implemented using a low pass filter on the data from the sensors to filter out data that might influence the result. Several algorithms were used such as the peak detection algorithm to help detect steps and the direction detection algorithm to calculate the direction the user is moving forward.

To improve the accuracy of localization, we also applied the KNN algorithm. Experiment results achieved an accuracy enhancement from 88% to 92%.

For the navigation part, the Fibonacci Heap was applied with the Dijkstra Algorithm as an optimization. The time complexity is reduces from from $\mathcal{O}(n^2)$ to $\mathcal{O}(m + n\log n)$. For a complex building with more than 500 doors in a floor, the runtime for the shortest path algorithm is reduced from 91.2 milliseconds to 2.5 millseconds.

References

[1] Pavel Davidson and Robert Piche. A survey of selected indoor positioning methods for smartphones. IEEE Communications Surveys & Tutorials, 19(2), 1347-1370, 2017.

[2] F. Zafari, A. Gkelias and K. K. Leung. A survey of indoor localization systems and technologies. IEEE Communications Surveys & Tutorials, 21(3), 2568-2599, 2019.

[3] Gabriel Deak, Kevin Curran, and Joan Condell. A survey of active and passive indoor localisation systems. Computer Communications, 35(16), 1939-1954, 2012.

[4] Hui Liu, Houshang Darabi, Pat Banerjee, and Jing Liu. Survey of wireless indoor positioning techniques and systems. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 37(6):1067-1080, 2007.

[5] Luca Mainetti, Luigi Patrono, and Ilaria Sergi. A survey on indoor positioning systems. 22nd International Conference on Software, Telecommunications and Computer Networks (SoftCOM), pages 111-120, 2014.

[6] K. Aamodt. Chipcon products from Texas Instruments. Application Note AN042 (Rev. 1.0), 2011. http://www.ti.com/product/CC2431

[7] Ralf Bill, Clemens Cap, Martin Kofahl, and Thomas Mundt. Indoor and outdoor positioning in mobile environments a review and some investigations on WLAN positioning. Geographic Information Sciences, 10(2):91-98, 2004.

[8] Suining He and Kang G Shin. Geomagnetism for smartphone-based indoor localization: Challenges, advances, and comparisons. ACM Computing Surveys (CSUR), 50(6):97, 2017.

[9] Robert Harle. A survey of indoor inertial positioning systems for pedestrians. IEEE Communications Surveys and Tutorials, 15(3), 1281-1293, 2013.

[10] T. Do-Xuan, V. Tran-Quang, T. Bui-Xuan and V. Vu-Thanh, Smartphone-based pedestrian dead reckoning and orientation as an indoor positioning system. International Conference on Advanced Technologies for Communications (ATC 2014), 2014.

[11] Valerie Renaudin, Christophe Combettes, and Francois Peyret. Quaternion based heading estimation with handheld MEMS in indoor environments. IEEE/ION Position, Location and Navigation Symposium, pages 645-656, 2014.

[12] Jeong Won Kim, Han Jin Jang, Dong-Hwan Hwang, and Chansik Park. A step, stride and heading determination for the pedestrian navigation system. Positioning, 1(08):0, 2004.

[13] Android Developers. Sensors Overview. https://developer.android.com/guide/topics/sensors

[14] Qi-wei Bao. An Indoor Oriented Localization and Navigation System. M.S. Thesis, Case Western Reserve University, Cleveland, Ohio. 2018.

[15] Edsger W Dijkstra. A note on two problems in connexion with graphs. Numerische mathematik, 1(1):269-271, 1959.

[16] Michael L Fredman and Robert Endre Tarjan. Fibonacci heaps and their uses in improved network optimization algorithms. Journal of the ACM(JACM), 34(3):596-615, 1987.