Indoor Localization using Smartphone Inertial Sensors

Yang Liu[1], Marzieh Dashti[2], Mohd Amiruddin Abd Rahman[1], Jie Zhang[1]
1: Department of Electronic and Electrical Engineering, University of Sheffield, Sheffield, UK
2: Bell Laboratories, Alcatel-Lucent, Dublin, Republic of Ireland
[1]{elu11yl, els11ma, jie.zhang}@sheffield.ac.uk
[2]marzieh.dashti@alcatel-lucent.com

***Abstract*—Celebrated fingerprinting techniques localize users by statistically learning the signal to location relations. However, collecting a lot of labelled data to train an accurate localization model is expensive and labour-intensive. In this paper, an economic and easy-to-deploy indoor localization model suitable for ubiquitous smartphone platforms is established. The method processes embedded inertial sensors readings through a inertial localization system. A particle filter is developed to integrate the building map constraints and inertial localization results to estimate user's location. To increase the algorithm convergence rate, the user's initial/on-line room-level localization is achieved using WiFi signals. To achieve room-level accuracy, only very few training WiFi data, i.e. one per room or per segment of a corridor, are required. A novel crowdsourcing technique to build and update training database is presented. On these basis, an indoor localization system is proposed and evaluated. The results show that comparable location accuracy to previous approaches without even dense wireless site survey requirements is achievable.**

I. Introduction

Among the state-of-art localization methods, the celebrated "fingerprinting" technique which utilize received signal strength (RSS) as a key metric to estimate user's locations shows to be quite successful. This method is based on statistically learning the signal to location relations [1, 5, 7, 13]. It is divided into two phases: training and positioning. Training phase involves a site survey process with a WiFi-enabled device, in which WiFi signal strengths from multiple access points (APs) at every position of an interesting area (RSS fingerprints) are recorded into a radio map (RM) database. Next in the positioning phase, when a user sends a location query with its current RSS reception, localization algorithms retrieve the RM database and return the matched fingerprints and corresponding locations.

The accuracy of fingerprinting technique is highly dependent on the density of fingerprints recorded in RM database. However, constructing a dense fingerprint database, with 1-2 meters fingerprint samples distance, is labour-intensive and not feasible. This work aims at reducing the burdensome survey effort while still offering accurate indoor positioning.

More and more people are relying on their smart phones to navigate them to their destinations. Taking advantages of embedded inertial sensors on ubiquitous smartphones yields an economic and easy-to-deploy indoor localization system.

Inertial localization has been originally utilized in rockets navigation [4]. The core device is inertial measurement units (IMUs), including compass, gyroscope and accelerometer. Since the IMUs have been packaged into small size by micro-electronic mechanical system (MEMS) [2], the MEMS-IMUs devices have become popular in smartphone nowadays. Ideally, with information from the IMUs and known initial position, user's motion can be tracked to update user's trajectory position. However, the results are vulnerable to small errors caused by movement variability, fix initial bias, etc [3, 10]. The filtering techniques can help to remove errors from individual sensor readings. Fusion schemes combine outputs of multiple sensors and floor plan informations to create improved results.

On these basis, an indoor localization system without dense wireless site survey requirements is developed in this paper. Embedded inertial sensors on smartphone are utilized to recognize the user's dynamic activities and walking directions. A computationally tractable and accurate novel real-time step detection algorithm using accelerometer is implemented as in [8]. Compass reading data is reported at proper moments to calculate the user's turning angle [6]. A particle filter is developed to integrate the building map constraints and inertial measurements. A robust algorithm to utilize floor map constraints to filter the impossible motion trends is presented.

To increase the algorithm convergence rate, the user's initial location is estimated with room-level accuracy, using WiFi signals [6]. The particles are initially distributed in the estimated initial room. To achieve room-level accuracy, only very few training WiFi data, i.e. one per room or per segment of a corridor, are required. Moreover, to increase convergence rate and to decrease computational load of particle filter, on-line room level localization algorithm is developed and applied to filter out the undesirable particles.

Wi-Fi infrastructures may be added/removed/ or moved to new places after building the fingerprints database. The un-updated RSS database causes inaccurate room-level localization. To keep the database updated, a simple crowdsourcing approach, so-called 'back-tracking' is proposed in this paper.

The detailed flow of the proposed positioning system is given in section II. And it is tested and evaluated in section III and IV. Finally section V summarizes the concluding remarks.

978-1-4799-4671-6/14/$31.00 ©2014 IEEE

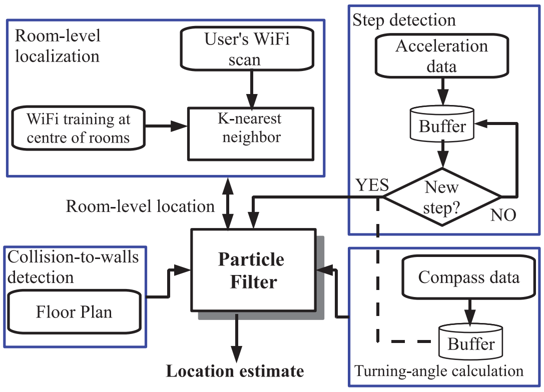

Figure 1: Diagram of the system

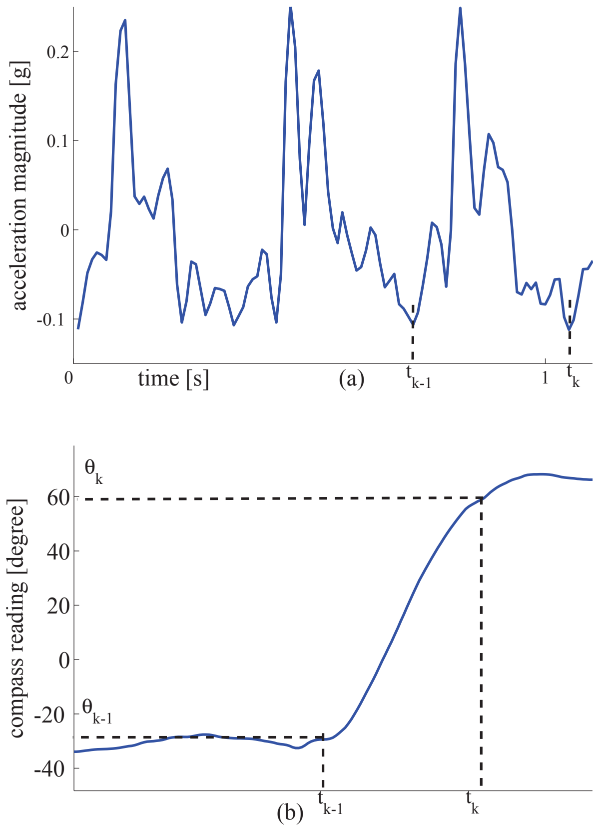

Figure 2: (a) iPhone built-in accelerometer recordings when user is taking three steps forward. When acceleration magnitude reached its local minimum, a step is completed. For instance, at time t_{k-1} and t_k, second and third steps are detected. (b) An example of compass reading when the user is turning after second step.

II. System Design

The overall vision of proposed positioning system is shown in Fig.1. The working process consists of four main phases:

1) localization with room-level accuracy using Wi-Fi signals.
2) step detection using accelerometer sensor.
3) turning angle calculation using digital compass sensor.
4) considering building floor plan constraints to avoid collision to the walls.

Finally the particle filter fuses this information to provide a robust tracking system. Particle filter employs particles to estimate user's position and motion trends. The key idea is that the particles should not occupy impossible positions given the map constraints and user's motion model. For example, since the user cannot pass through walls, the particles that cross a wall are weighted down. When a new step is detected, user's motion model is updated by compass readings. Finally, the user's position is estimated to be the average of particles positions. Note that initially these particles are distributed in a limited range based on the user's WiFi reception data. High level architecture and details are described in following subsections.

A. Analyzing accelerometer and compass data

Due to repetitive nature of walks, when a user walks continually, similar pattern for every step is found on accelerometer reading, as shown in Fig. 2 (a). In [8], a step detection algorithm based on computing the cross-correlation of the previous sequence of accelerometer readings samples with current sequence of accelerometer readings samples on the direction of walks is presented. In this system, similar approach is utilized. Every time a new step is detected, particle filter estimates the new position status by combining building floor map constraints information and compass measurements. The compass reading samples are stored in buffer and processed at the right time, i.e., the moments that ever step is detected.

Fig. 2 (b) shows an example of compass reading when user is turning. Steps are detected at time t_{k-1} and t_k. Hence, compass data at time t_{k-1} and t_k are taken. The difference between compass readings at time t_{k-1} and t_k gives the user's turning angle after second step.

B. Particle filter

Particle filter is a practical implementation of recursive Bayesian filter with the sequential Monte Carlo methods. Bayesian filter estimates a system state vector by constructing the posterior probability density function (pdf) of that state [11]. The posterior pdf should reflect all the up-to-date knowledge about the system state vector. When new information is received by recursive Bayesian filter, propagating and updating of the previous posterior pdf will be implemented. Particle filter approaches probability density function by distributing different weights to large amount of randomly generated samples. All the randomly generated samples are propagated and updated with the motion model and measurement model, respectively. Unlike Kalman filter, particle filter does not requires a prior knowledge of the exact position of user. Also, the model of the inherent noise embedded in newly entered information and measurement used by particle filter is not limited to Gaussian distribution [9].

The particle filter implemented in this system and the assumption used are similar to [6]. The difference is the mechanism of filtering particles. Convergence rate of particle filter in this system is faster and reliable as well. The contribution is from implementing on-line room-level localization technique. More details will be explained later in this paper.

C. Map constraints

Building floor map constraints are considered to calculate possible intersections of particles' trajectory with walls. A collision between moving particle and wall represents that a possible user moving path intersects walls. As a result, the particle should be filtered.

In this paper, an easy-to-implement collision detection algorithm is introduced. Given the floor plan of the building, the coordinates of every walls and partitions are recorded in the cloud server. To determine if the user's trajectory collides a specific wall or not, a novel approach is applied as following: Given both the coordinates of user's current location, (x_1,y_1), and possible next location,(x_2,y_2) as motion endpoint's coordinates, a linear equation representing the user's trajectory can be written out , in the form of $y = kx + b$. Similarly, linear equations for wall vectors could be built like that as well. Then, by substituting two motion endpoint's 2-D coordinates for x and y in wall linear function, two values of M_1 and M_2 are obtained. Similarly, by substituting two wall endpoint's coordinates for x and y in linear motion function, two values of W_1 and W_2 are obtained. Three different scenarios may happen:

- M_1 and M_2 have the same polarities (they are both negative or both positive values) AND the polarities of W_1 and W_2 are the same as well: in this scenario user's trajectory does not collide the wall. See black dashed line shown in Fig. 3.
- M_1 and M_2 have different polarities (one of them is a negative value and the other is positive) AND W_1 and W_2 have different polarities as well: in this scenario user's trajectory collide the wall. See blue dashed line shown in Fig. 3.
- either { M_1 and M_2 have different polarities AND W_1 and W_2 have same polarities } OR { M_1 and M_2 have same polarities AND W_1 and W_2 have different polarities } : in this scenario user's trajectory does not collide the wall. See red line shown in Fig. 3.

D. Room-level localization

1) Initial room: In the offline phase, few Wi-Fi training data (RSS fingerprints) are collected only at the center of the rooms, and center of each segments of long corridors. These few training data limits the initial range of particles. Consequently the filter convergence to the exact user's location more quickly. The basic idea is that the user's online Wi-Fi scans are compared to all few training data to find the nearest training data in signal space. In this system, we use the algorithm presented in [12] to get the comparison results.

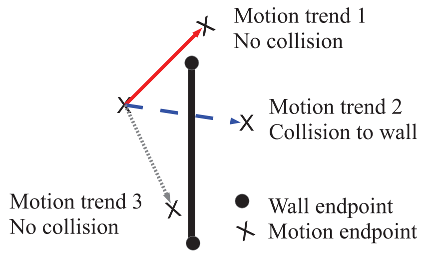

Figure 3: A collision is detected in the second scenario.

That gives an initial room level accuracy of the user's position. Then, particles are uniformly distributed in and only in the user's estimated initial room [6].

2) On-line room-level localization: As it is described, particle filter fuses inertial data and map constraints and accordingly updates the particles continuously. At every step, the user's location is calculated as the average of survived particle's positions.

Often when users go out from a room, experience a turn at corridor and then go along the corridor, survived particles will be distributed in several different room or corridor segments, so called 'multi-clustered particles'. Clustered particles far from ground true position of user, as shown in Fig. 7, have serious negative impact on location estimate.

To address this problem, on-line room-level localization algorithm is applied. This algorithm only runs when a dramatic change in user's status is observed, e.g., when compass readings at two consecutive steps vary significantly.

The algorithm aims at limiting the particles in one room or one segment of a corridor. This task is also finished by comparing on-line Wi-Fi scans with training fingerprints. The room or corridor segment holding the smallest difference in signal space will be selected to be on-line room localization result. The particles beyond determined room or corridor segment range are filtered out. It should be noted that these useless particles are always filtered when their future movements intersects with walls even if on-line room-level localization technique is not used. In other words, the application of on-line room-level localization technique accelerates the process of dismissing these particles, which saves the computational load involved in propagation of and collision detection of useless particles as well. Moreover, on-line user location estimation error is further reduced by filtering these particles by on-line room-level localization technique.

E. Updating the database through crowd-sourcing

Wi-Fi infrastructures may be added/removed/ or moved to new places according to wireless coverage requirements. The un-updated RSS database causes inaccurate room-level positioning. Also, Collecting signal fingerprints deliberately to build a training database covering all points on the interested indoor area is a time-consuming effort. For tackling this problem, a simple crowd-sourcing approach, so-called 'backtracking' is developed to keep the database updated.

To implement the back-tracking approach, all particles are labelled uniquely at the start point of the trajectory. The number of survived particles reduces by time due to filtering mechanism from on-line room-level localization and collision detection. Hence at the end point of the trajectory, the final survived particles are more exactly converge to the exact user's position. The algorithm track the survived (and labelled) particles backward from the end of the trajectory to the start point. Trough back propagation of final survived particles, the position information of these tagged particles throughout the whole trajectory are obtained. Average of particles' position at every step is calculated as user's location, which will be shown later in this paper that it is very close to the user's real location.

The back-tracking algorithm is applied in the off-line analysis phase after users arrive at the desired locations. In this way, the Wi-Fi fingerprints database covering whole interested indoor space could be built after several hours trial and updated as well. Then, on-line localization accuracy could be benefited from comparing on-line Wi-Fi scans with the fingerprints data in the database.

III. Simulations

For evaluating the performance of the designed system, an indoor test environment is simulated. Fig. 4 shows the test environment layout. Four WiFi APs are deployed in corners of the four corner rooms. Wi-Fi data is simulated using realistic simulator, Ranplan iBuildNet©. The simulator comprises of building modelling (structure and material property) and radio propagation tools. Also, in the above figure, the signal coverage of the Wi-Fi AP installed at room 2, at upper right corner of building is shown. RSS training vector only at the center of each room and center of the corridor segments are measured and recorded.

Assume that a user holding iPhone sits in the room 1 (at the lower left corner of building) at first time stamp. The user then stands up from the seat and moves to room 2, walking on the path shown in Fig. 9 (a). Especially, we assume the doors of all rooms could be opened and got through by user. It is believe that this assumption is reasonable because localization system do not have prior knowledge of which rooms is accessible to users. Thus, particles are assumed to be able to get through all doors freely. The user's smartphone is collecting accelerometer and compass data during whole movements. Fig. 5 shows the user's acceleration fluctuation during standing up and first steps afterwards. The on-line RSS vector data on the user's walking path are also measured by iBuildNet©.

IV. Results

A. convergence results on particle filtering

By comparing the dissimilarity between the user's initial on-line RSS vector and training RSS vectors, the algorithm estimates the user's initial position to be in room 1, hence, the initial particles are uniformly distributed in room 1.

Particle filter is updating the particles through the motion model continuously. Fig. 6 (a) shows examples of the survived particles at three different time stamps, at the beginning, middle and last part of the walking path. The number of survived particles, i.e., user's possible positions, is decreased over time. Algorithm converges to the exact position of the user after taking enough number of steps.

As it is described, when a dramatic change in user status occurs, e.g. turning near border of room and corridor, the survived particles will be distributed in several different clusters (room or corridor segments). Fig. 7 shows an example of such this scenario. Here, by applying the on-line room-level localization, a cluster of undesirable particles are filtered out. Hence not only the computational load of particle filter is decreased but also the algorithm converges to exact user's location more quickly. Fig. 8 compares the convergence rate of the particles with and without applying the on-line room-level localization algorithm.

B. positioning results

Fig. 9 compares the real position and estimated position by our designed system through the whole walking path. The mean value of positioning error is 62 cm. The positioning error statistics are summarized in Table I. At the start point of walking path, only room-level accuracy is achieved. The

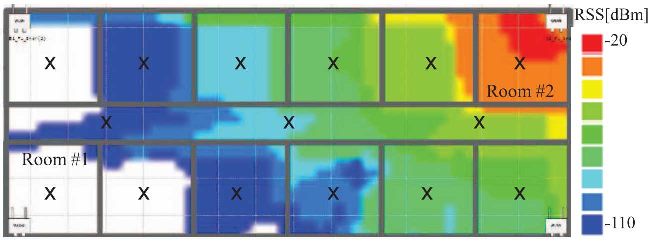

Figure 4: Predicting Wi-Fi signal coverage within the building using Ranplan $iBuildNet^{©}$ radio propagation tool. RSS Wi-Fi fingerprints are collected only at center of the rooms and center of the corridor segments.

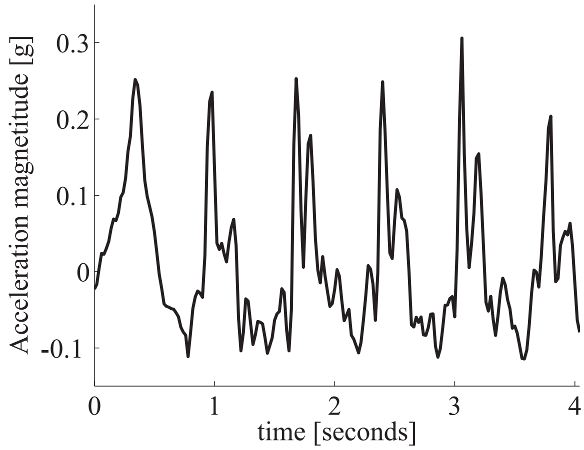

Figure 5: Acceleration magnitude when user standing up from the seat and takes steps forward.

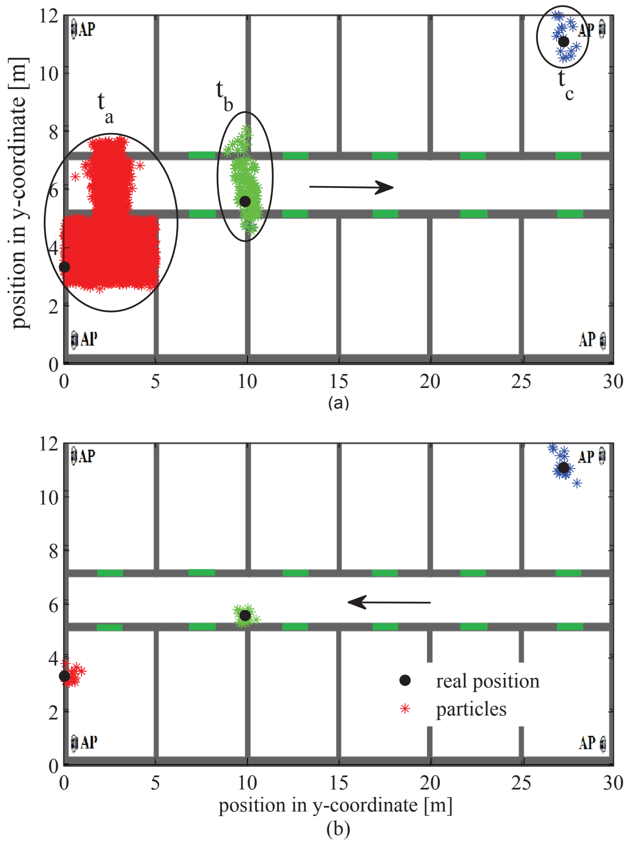

Figure 6: Example of survived particles at three different time-stamps. User's real position is marked with black circle. (a)online positioning (b)back-tracking

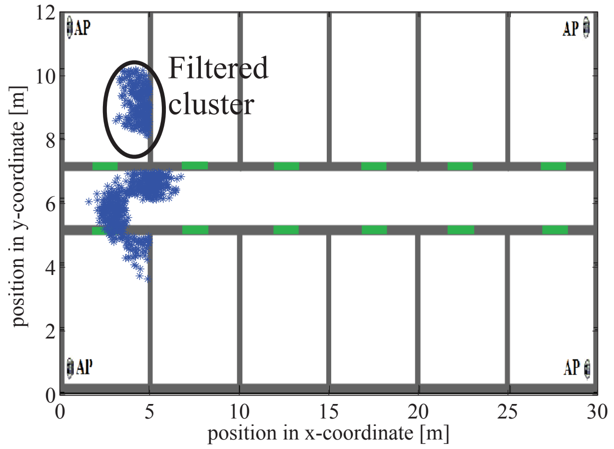

Figure 7: The particles are distributed in multi-clusters. Room-level localization algorithm filtered out one cluster of undesirable particles.

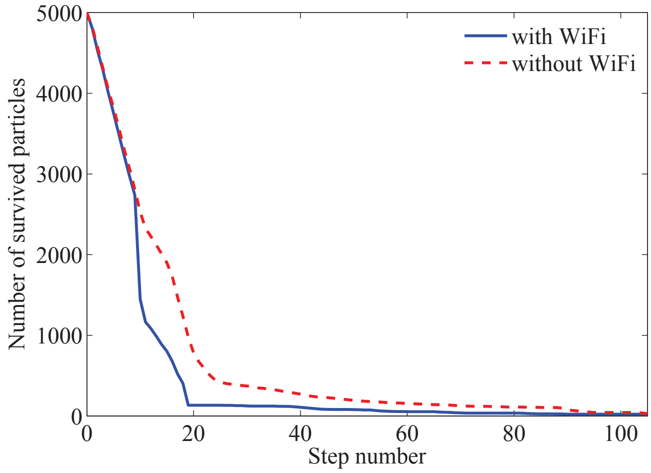

Figure 8: Reduction rate of survived particles by time, with and without applying on-line room-level localization technique.

Table I: Statistics of positioning error, with and without applying on-line room-level localization technique

error statistics	with WiFi	without WiFi
mean [m]	0.62	1.33
standard deviation [m]	0.65	0.50
maximum error[m]	2.61	2.58
minimum error [m]	0.01	0.29
error at first position[m]	2.61	2.58
error at last position[m]	0.42	1.22

positioning error is gradually reduced from 2.61 m to 0.42 m from the start point of the walking path to the end point. Fig. 10 shows the reduction of ranging error by time.

Fig. 10 also, further, depicts that applying on-line room-level localization, marked by blue line, to filter out undesirable particles can improve the positioning system accuracy, compared with scenario that does not apply that technique (marked by red dashed line).

C. *back-tracking result*

At the end of the user's trajectory, the number of final survived particles is 29. The historical movements of these particles have been tracked backward. For example, the positions of them during three time-stamps are shown in the Fig. 6 (b). The average of the positions of these particles are calculated at every time-stamp. Also, in the Fig.9 (b), the tracked path of these particles is compared with the real user's walking path. As it is observed clearly in Fig. 9 (a) and (b), the backward analysis path is more closely matched to the real path than the on-line localization result. Hence, in the offline phase, the information from the backward analysis path, i.e., position and RSS data, is used to update the fingerprints database.

V. Concluding Remarks

This paper presented an economic and reliable indoor positioning model suitable for smartphone platforms. Iner-

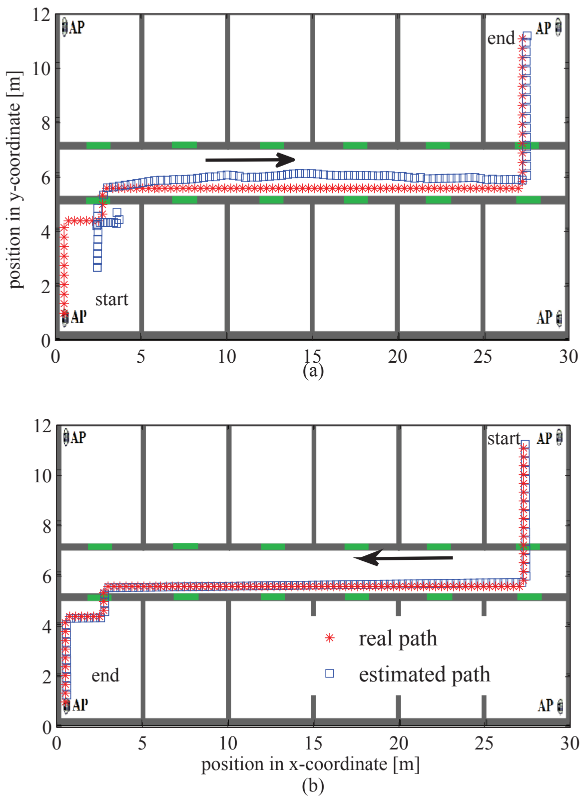

Figure 9: Real path vs. estimated path (a) on-line positioning (b)back-tracking

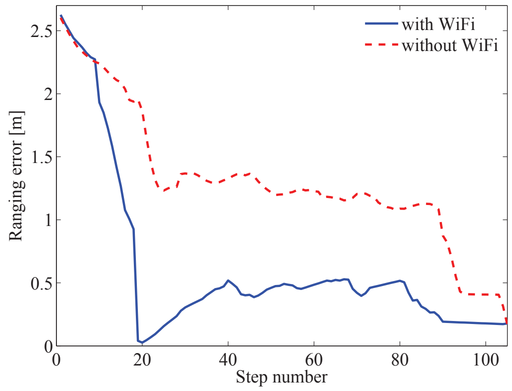

Figure 10: Positioning error with and without applying on-line room-level localization technique.

tial localization algorithm involving analyse of data from accelerometer and digital compass has been implemented. A easy-to-implement collision detection algorithm has been developed, which activates usage of environment floor map to detect useless particles. Particle filter integrates inertial data and building map constraints to localize users. Only very few Wi-Fi training fingerprints for initial/on-line room-level localization and crowdsourcing approach are required , hence burdensome effort of building training fingerprints database required in common fingerprints positioning system is avoided and convergence rate of particle filter has been improved a lot. Furthermore, crowdsourcing approach is also function of updating the database. Comparable positioning accuracy (mean error of 1.33 m for on-line localization in a 30m $\times$ 12m building) to previous approaches without even dense wireless site survey requirements is achieved.

References

[1] P. Bahl and V.N. Padmanabhan. Radar: An in-building rf-based user location and tracking system. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 775–784. IEEE, 2000.

[2] A. Brown and Y. Lu. Performance test results of an integrated gps/mems inertial navigation package. In *Proceedings of ION GNSS*, pages 21–24, 2004.

[3] C. Jekeli. *Inertial navigation systems with geodetic applications*. de Gruyter, 2000.

[4] AD King. Inertial navigation-forty years of evolution. *GEC review*, 13(3):140–149, 1998.

[5] A.M. Ladd, K.E. Bekris, A. Rudys, L.E. Kavraki, and D.S. Wallach. Robotics-based location sensing using wireless ethernet. *Wireless Networks*, 11(1-2):189–204, 2005.

[6] Y. Liu, M. Dashti, and J. Zhang. Indoor localization on mobile phone platforms using embedded inertial sensors. In *Proceedings of the 10th Workshop on Positioning, Navigation and Communication*, pages 1–5, 2013.

[7] V. Otsason, A. Varshavsky, A. LaMarca, and E. De Lara. Accurate gsm indoor localization. *UbiComp 2005: Ubiquitous Computing*, pages 903–903, 2005.

[8] A. Rai, K.K. Chintalapudi, V.N. Padmanabhan, and R. Sen. Zee: Zero-effort crowdsourcing for indoor localization. In *Proceedings of the 18th annual international conference on Mobile computing and networking*, pages 293–304. ACM, 2012.

[9] B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman filter: Particle filters for tracking applications*. Artech House Publishers, 2004.

[10] I. Skog and P. Handel. In-car positioning and navigation technologiesï¿$\frac{1}{2}$a survey. *Intelligent Transportation Systems, IEEE Transactions on*, 10(1):4–21, 2009.

[11] O. Woodman and R. Harle. Pedestrian localisation for indoor environments. In *Proceedings of the 10th international conference on Ubiquitous computing*, pages 114–123. ACM, 2008.

[12] C. Wu, Z. Yang, Y. Liu, and W. Xi. Will: Wireless indoor localization without site survey. In *INFOCOM, 2012 Proceedings IEEE*, pages 64–72. IEEE, 2012.

[13] M.A. Youssef, A. Agrawala, and A. Udaya Shankar. Wlan location determination via clustering and probability distributions. In *Pervasive Computing and Communications, 2003.(PerCom 2003). Proceedings of the First IEEE International Conference on*, pages 143–150. IEEE, 2003.