EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

# SmartLoc: sensing landmarks silently for smartphone-based metropolitan localization

Cheng Bo[1], Taeho Jung[2], Xufei Mao[3], Xiang-Yang Li[4,2*] and Yu Wang[1,5*]

**Abstract**

We present *SmartLoc*, a system to estimate the location and the traveling distance of a vehicle in urban environments by leveraging the lower-power inertial sensors (e.g., accelerometer, gyroscope, compass) embedded in smartphones. It has been well documented that the inherent noise in the data measured from these sensors will result in a huge drift in the final estimated moving distance and hurt the performance of location-based applications. In *SmartLoc*, we designed, tested, and evaluated a sequence of techniques to address this notoriously challenging task of noise canceling. By exploiting the intermittent strong GPS signals, we use the linear regression to build a prediction model from the trace estimated from inertial sensors and the trace computed from GPS. We extensively rely on automatically detected landmarks (e.g., bridge, traffic lights) and special driving patterns (e.g., turning, uphill, and downhill) from inertial sensors data to improve the localization accuracy when the GPS signal is weak. Our evaluations of *SmartLoc* in a large city demonstrate its technique viability and significant localization accuracy improvement compared with GPS and other approaches: the error is approximately within 20 m for 90 % of the time. *SmartLoc* system can also reduce the energy consumption for localization by carefully turning off GPS under certain scenarios.

**Keywords:** SmartLoc, Inertial sensors, Localization, Smartphone, Mobile big data

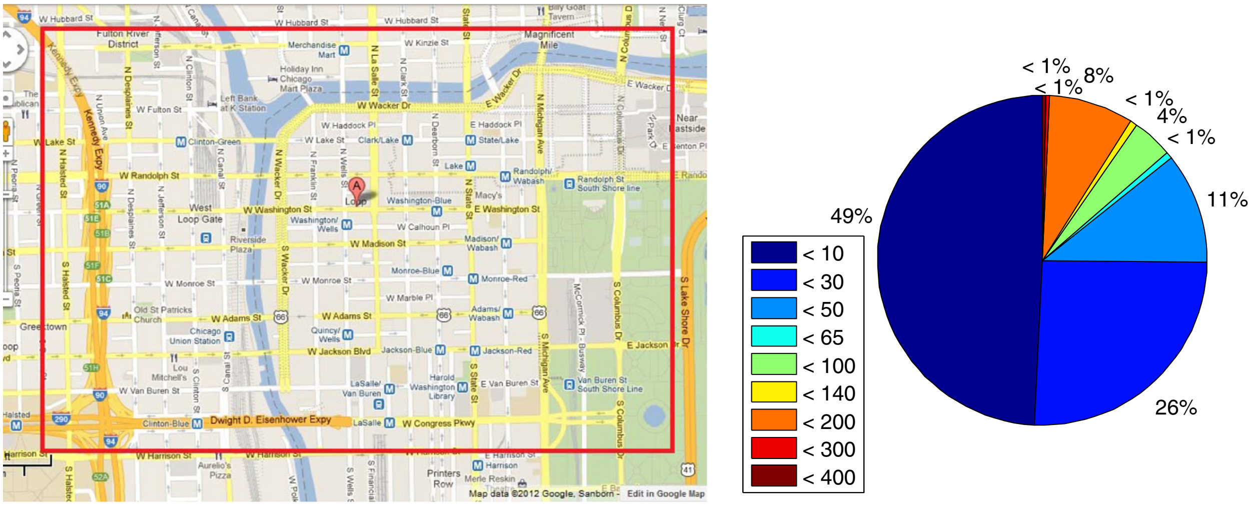
## 1 Introduction

Localization has attracted significant attentions in the past few decades (especially with the emerging smart city applications), and numerous techniques have been proposed to achieve high-accuracy localization. Although GPS is the most common technology to provide accurate location for outdoor environment [1–6], it still suffers from signal blocking and multipath effect, which lead to significant low location accuracy in metropolises. On the other hand, many smart city and cyber-physical system (CPS) applications and systems (such as vehicular networks [7, 8] and urban crowd sensing [9, 10]) heavily rely on accuracy location information. Therefore, it is crucial to investigate new urban localization techniques beyond GPS.

To study the severity of the GPS localization errors in metropolises, we collect the GPS information (location and the location accuracy) in all road segments in downtown Chicago through the integrated GPS module in smartphones. Our measurement (detailed results in Section 3) demonstrates that, due to highrises along two sides of the roads, the GPS signals are often very weak and unstable, and in some complicated road segments (e.g., tunnels and underground), the GPS signals are totally blocked. Under these circumstances, the location accuracy is extremely poor, e.g., the largest error in our measurement is over 100 m on the ground, and nearly 400 m in the underground segments (see Fig. 1 for more details). Therefore, improving the location accuracy is imperative when GPS signal is weak in metropolises.

A number of novel techniques have been proposed in the literature to remedy the inaccuracy of GPS localization in metropolises [1, 11–13] and to reduce the energy consumption of GPS process [14, 15] or time complexity [16]. However, such methods often require pre-established fingerprint databases, which are costly to collect, and looking-up such large-scale databases is also computationally intensive. In this work, we leverage minting the mobile data from existing sensors (e.g.,

*Correspondence: xli@cs.iit.edu; yu.wang@uncc.edu
[4] School of Computer Science and Technology, University of Science and Technology of China, Huang Shan Road, Hefei, Anhui 230027, China
[1] Department of Computer Science, University of North Carolina at Charlotte, 9201 University City Blvd., Charlotte, NC 28223, USA
Full list of author information is available at the end of the article



(a) Downtown of Chicago (b) Proportion of Errors

**Fig. 1** GPS localization accuracy in the downtown area of Chicago. **a** A map of the test region in the downtown of Chicago. **b** The proportion of different road segments with certain GPS localization errors (in unit of meters)

GPS, accelerometer, gyroscope, and magnetometer sensors) that are already integrated in most smartphones to provide more accurate localization for vehicular users in urban environments.

Exploiting the data collected from these sensors has been used in the literature to address a number of challenging and interesting tasks, e.g., indoor localization [17–20], road condition monitoring [21, 22], property tracking [13], user security [23, 24] activity recognition [25–28], measuring walking speed [17], and outdoor localization [13, 28, 29]. In contrast to these novel solutions, the main challenges in estimating the location (or trajectory) using inertial sensors include:

1) The inherent noise, inaccuracy, and imprecision in the sensory data [13, 18] cause significant drift (from cumulative errors) in estimated location, or trajectory even the smartphone is fixed inside the vehicle.
2) The coordinating systems on the smartphone (called body frame coordinate (BFC)) and earth (called earth frame coordinate (EFC)) are different. The conversion of the acceleration data from BFC to EFC will introduce additional noise to the data as we can only estimate the smartphone's orientation.
3) The smartphone's dynamically rotating BFC (caused by the movement of smartphone) introduces additional errors in location estimation.[1] We notice in our testing that even if we mount the smartphone steady, its orientation changes slightly but still irregularly. It is thus extremely challenging to convert the acceleration data to EFC while minimizing the introduced errors.
4) It is difficult to distinguish whether the vehicle is driving at a constant speed or is stopped as in both scenarios, theoretically, the acceleration speed is around zero.

Although motion sensors have been used in pedestrian navigation, we argue that locating a vehicle in metropolises is far more challenging. For pedestrian navigation, we can count the number of steps based on the cyclic pattern responding to walking steps and then estimate the distance using step length and the direction using the compass. In contrast, car-driving does not have such a nice cyclic pattern for the collected accelerometer data. In this paper, we design a smartphone-based localization prototype, SmartLoc, based on Android smartphones, and evaluate its performance both in the downtown of Chicago and on highways. Our extensive real-world experimental testing results indicate that SmartLoc significantly improves the location accuracy compared with GPS data in downtown areas. When evaluated in Chicago downtown, the localization error is at most 20 m for 90 % of the locations. Our measurement shows that for $\geq$ 50 % of the locations, GPS has a localization error more than 20 m. To support our design of SmartLoc, we first conduct extensive measurement of the GPS localization accuracy in the majority of the blocks in Chicago, pointing out the factors influencing the location accuracy in a metropolis. In addition, when testing on highways, the localization error is at most 12 m for 95 % of the time or about 2 % of the driving distance. This result also encourages energy saving by turning off the GPS when the GPS signal is weak.

We also analyze the characteristic and performance of integrated sensors in the smartphone in detail, and investigate the effect brought about by the sensors in real localization tasks. We propose a sophisticated mechanism to calculate and optimize the continuous orientation of a smartphone and estimate the location while driving. We also design a calibration strategy based on the road infrastructures (e.g., bridge, traffic lights, uphill, and downhill), and driving status (e.g., turns, stops), which are inferred from the sensor data, such as accelerometer, gyroscope, and magnetometer. In AutoWitness [13], the error of distance estimation is less than 10 % for most of the cases, which could be large enough when the estimated distance is long (10 % of the 2 miles is 0.2 miles). In comparison, the result of SmartLoc indicates that, in 90 % of all cases, estimation error is less than 20 m, and in the highway, the error is even lower.

The rest of the paper is organized as follows. In Section 2, we briefly review the state-of-the-art localization techniques. We then provide our measurement results and observations on the GPS accuracies and volatile fluctuations of the sensory data via field experiments in Section 3. In Section 4, we present the overview of our system. We then provide our detailed localization solutions using MEMS sensors and how to deal with noisy data in Section 5. In Section 6, we discuss how the road infrastructures and driving features are used to improve the localization accuracy. We report our extensive real-world experimental results in Section 7 and conclude the work in Section 8. A preliminary version of this work appeared as a poster [30] in ACM MobiCom 13.

## 2 Related works

Our work intersects with several localization techniques, and in this section, we mainly focus on the review of wireless localization and dead-reckoning [31].

### 2.1 Wireless localization

The most popular outdoor localization method is GPS [14]. In an open environment, GPS could provide extremely accurate location information to the user and more improved techniques such as WAAS [32], and differential GPS (DGPS) [33] have been adopted in general civilian devices. Another feasible method is GSM-based localization [12], such as Nericell [21], which is widely available but with low accuracy (up to hundreds of meters). Also, it needs to know the exact position of cellular towers in advance.

Other techniques for both outdoor and indoor localization mainly fall into two categories: *fingerprint-based* and *modeling-based*. The main idea of the fingerprint is to first establish a database of signal fingerprint of each location based on the surrounding environment and then match the measured fingerprint to the database. The mapping returns the location with the most similar fingerprint. Modeling-based techniques rely on some theoretical or experimental signal propagation models, relating delay or strength to distance, and then calculate the location based on estimated distances.

In an indoor environment, Radar [34] collects WiFi fingerprints beforehand at known locations inside a building and identifies the user's location using the matching fingerprint. It could achieve relatively high accuracy, with a median error of 2.94 m. Similar systems using RF signals include Horus [35] and LANDMARC [36]. Fingerprints generated through the combination of WiFi and FM are also studied in [37], which increases the accuracy by 83 % compared with WiFi fingerprint only. Unlike other systems mentioned above, SurroundSense [38] estimates the location in indoor environment through multiple ambience features, including the light, the color of the floor, the WiFi, and the sound. LiFS [19] does not have to know the APs' location in prior, and the high location accuracy is achieved through exploiting user's motion from crowdsourcing. PinLoc [39] uses physical layer information to locate users with the accuracy $\sim$ 89 %.

In an outdoor environment, both PlaceLab [1] and ActiveCampus [40] make full use of both WiFi and GSM signal for location. The former creates a map by war-driving a region and maps both APs and cell towers' signal against the wireless map. The latter is quite similar except it assumes that the APs' location is known prior. Taking advantages of both systems, CompAcc [17] uses dead-reckoning combined with AGPS to calibrate rather than preliminary war-driving. All these systems demand time-consuming calibration and are not suitable for a large-scale area, such as urban environments. StarTrack [41] provides a framework for track-based applications, which does not require precise location. However, our work mainly focuses on providing accurate real-time location in outdoor environments, which provides fundamental techniques for location-based service and navigation.

Other common ways to calculate location are based on the geometric model. In time-based techniques, time of arrival (ToA) [42], time difference of arrival (TDoA) [43], and angle of arrival (AoA) [44, 45] establish a geometric relationship between two separate devices and calculate the location. Although pleasing accuracy is achieved, the high cost cannot be neglected. In acoustic ranging, the distance of transmitter and receiver is estimated more precisely. Existing work includes Active Bat [46], Cricket [43], ENSBox [47], and WALRUS [48].

Liu et al. [20] calibrate the location accuracy in the WiFi fingerprint using a more accurate distance estimation from acoustic ranging. Centaur [49] combined acoustic ranging and RF measurement to locate a device in an office environment with acceptable accuracy. However, the acoustic approach is not suitable for large-scale localization either.

Several promising techniques such as crowdsourcing are introduced in localization recently, such as Zee [50], which also uses inertial sensors to track users' movement. However, this inertial sensor-based approach is only suitable for indoor localization, and using such sensors for outdoor localization is more challenging.

### 2.2 Dead-reckoning

Recently, dead-reckoning using internal sensors to estimate motion activities attracts a lot of research interests. For example, Strapdown Inertial Navigation System (SINS) [51] and Pedometer System [52] use MEMS to estimate the moving speed and trace. The key issue is to deal with noise of internal sensors, and the accumulated errors, which sometimes grow cubically [53]. Personal Dead-reckoning (PDR) system [11] uses "zero velocity update" to calibrate the drift. The majority of the dead-reckoning results focus on walking estimation, such as UnLoc [18] and CompAcc [17]. They mainly use an accelerometer to estimate the number of steps and then measure the distance through estimated step length. AutoWitness [13] is a system with an embedded wireless tag integrated with vibration sensors, accelerometer, and gyroscope. The tag is attached to a vehicle, and accelerometer and gyroscope are used to track the moving trace.

## 3 GPS accuracy measurements in urban environments

To study the accuracy of localization via current GPS devices in urban environments, we first conduct comprehensive measurements of GPS accuracies within the red rectangular area in the downtown of Chicago, as shown in Fig. 1a. We find that in this test region, the largest GPS location error reaches 400 m and the length of the longest road segment with reasonable accuracy ($\leq$ 30 m) is about 1 km.

### 3.1 Measurement and experiment settings

We dispatch four cars, driving through every road segment in the area, including the road infrastructure such as bridge and underground tunnels. While driving, we record the longitude, latitude, location accuracy, and driving speed in every 50 ms. The GPS location accuracy is provided in the GPS data collected from the smartphone. Meanwhile, we also periodically (20 Hz) collect all the sensory data coming from motion sensors, which will be used to identify the driving condition. To remove time-dependent GPS location errors, we conducted three independent measurements at three different times. The results reported here are the average of these three measurements.

### 3.2 Localization results based on GPS signals

Our measurements generated over 100,000 sample points during driving. The location accuracy was not so high as expected. The smallest error we collected from GPS was 5 m, while the largest error reached 400 m, which is nearly the length of three blocks downtown. Such a high error will lead to the wrong instruction for turning or stopping in the navigation.

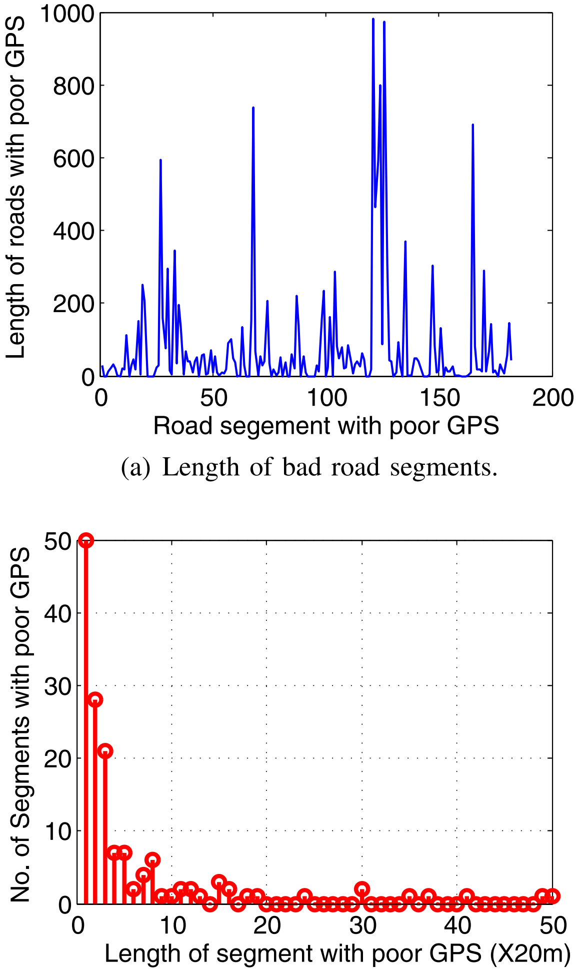
We also calculated the proportion of different road segments with certain GPS localization error and plotted the results in Fig. 1b. From this figure, we could find that only about half of the sampling points endure the error of less than 20 m, over one quarter of the locations have an error of about 50 m, and the rest of the quarter has an error larger than 50 m.

Localization is a fundamental step for navigation and location-based services. We assume that, while driving in metropolis, the largest location error we could accept should be less than a quarter of one block, which is about 30 m. For simplicity, we denote the position with GPS location error less than 30 m as the place with *good GPS signal*, and the rest as the position with *poor GPS signal*.

Our measurement shows that $\geq$ 25 % of sampling points have low GPS localization accuracy (i.e., with poor GPS signal). However, this does not imply that GPS cannot provide good navigation services. Typically, only under the circumstance that GPS localization accuracy is low for a whole segment of roads will the navigation system work poorly. For this reason, we then examined the length of continuous road segments that do not have good GPS signal, and we denote such road segment as "bad segment." We find from our measurement that there are 182 straight bad segments of roads, see Fig. 2a for some statistical information. The length of the longest bad road segments are almost 1 km, and the average length of all bad road segments is approximately 200 m, which is nearly two blocks. We also plot the number of bad segments in each length in Fig. 2b. From this figure, we observe that there are about 9 bad segments with length over 400 m, which are all located in the center of downtown.

## 4 System overview of SmartLoc

In this section, we present the overview of the SmartLoc architecture. The main idea is to use the smartphone to monitor the motion condition of vehicle and provide precise location, especially when the GPS signal is weak. There are several design challenges in order to meet such requirements in dynamic urban environments, while balancing between energy consumption and localization accuracy. (1) The first challenge is which sensors should be chosen and how to apply sensor fusion. (2) Accumulated errors will be imported when estimating traveling distance through tradition Newton's laws of motion. How to handle such errors is a challenging task. (3) The



(a) Length of bad road segments.

(b) Number of bad road segments.

**Fig. 2** Road segments with poor GPS signals

**Algorithm 1:** Overall Architecture of SmartLoc

```
while Driving do
    Collecting sensing data from GPS,
    accelerometer, gyroscope, magnetometer
    within the smartphone;
    Removing noises from raw data (e.g., slide
    window);
    Computing smartphone orientation (e.g.,
    Extended Kalman Filter);
    foreach time slot do
        if GPS signal is good then
            Building a dead reckoning predication
            model using GPS data and other
            inertial sensors (e.g., Linear
            Regression);
            Updating location using estimated
            speed and moving distance;
        end if
        else
            Estimating traveling distance using
            built predication model;
            Detecting if there is any special
            landmark or driving state;
            if Landmark or turn or traffic light
            detected then
                Calibrating the location using
                map information;
            end if
            Update location using estimated
            distance and calibration adjustment;
        end if
    end foreach
end while
```

third challenge is to detect and recognize the landmarks which will be used to improve the localization accuracy. Algorithm 1 presents the overall architecture of our system.

Usually, the smartphone is mounted on the windshield during navigation; the activity of vehicle is reflected though the sensory data. Since the large amount of error inherited from orientation estimation leads to poor traveling distance calculation when driving, we build a moving distance prediction model by training the distance estimation model through linear regression to minimize the accumulated error. The localization will be conducted by timeslots, and the parameters generated from linear regression will be introduced into the distance prediction for next timeslots. Due to the complex road condition and unpredictable driving behavior, the training process should be updated periodically. In addition, SmartLoc will detect the landmark by finding a specific pattern of the collected data sequence and calibrate the location accordingly. In the following three sections, we will introduce the detailed methods of our proposed system, including how to collect sensing data, remove noises, estimate orientation, calculate distance and position, build the predication model, detect landmark, and calibrate the location.

## 5 Locating vehicle basics

In this section, we mainly focus on estimating the moving velocity and distance of the vehicle using the motion sensors in smartphones. However, aligning both the orientation of smartphone to the earth coordinate and the accumulated error generated from linear acceleration, the traditional method of both velocity and distance estimation will contain increasing deviation to the ground truth. Under this circumstance, it still cannot provide precise position information to the driver. In the following paragraphs, we will discuss both the localization mechanism and the noise cancelation in detail.

### 5.1 Characteristics of sensory data

Since the behavior of the accelerometer acts like a damped mass on a string, the sensor in the device measures not only the linear acceleration applied to the device but also the force of gravity. In addition, the intrinsic noise of the motion sensors also affect the orientation estimation of the smartphone, which also leads to dramatic velocity and distance deviation eventually. To verify our assumption, we analyze noises in accelerometer with a set of experiments by putting the smartphone on a horizontal ground in different orientations. By transforming the acceleration from the device's coordinate system to the Earth Frame Coordinate system, we plot both the mean value and standard deviation of acceleration in different axes in Fig. 3a, b, respectively. From these two figures, it is obvious that the mean is not zero even when the phone is static, the readings of acceleration in three axes fluctuate frequently over the time domain, and the errors also change in different orientations.

In addition, the readings from magnetometer in the smartphone is sensitive to ambient environments, especially around a vehicle. We collect the magnetic field in three dimensions in two different scenarios: in and out of car, and calculate the mean value and standard deviation for three axes in different cases, illustrated in Fig. 3c, d. From these figures, it is obvious that even if the smartphone is in the same position, the magnetic field readings differ a lot. Meanwhile, both means and standard deviations in the car are approximately over 100 times larger than that outside a car, which means even if the smartphone is stationary, the fluctuation is larger while driving. The key influence comes from the engine.

However, the performance of gyroscope is much stable than the others as can be seen in Fig. 3e, f. These readings from all three scenarios are similar, which provide us a good opportunity to improve the device's orientation estimation and the following trajectory calculation.

### 5.2 Advanced orientation estimations

Orientation of the smartphone is significant in the process of trajectory estimation. However, traditional algorithms calculate the orientation with respect to the earth though the combination of accelerometer and magnetometer, which will introduce inevitable large errors. However, the gyroscope could compensate such a shortage. In this stage, we will present in detail our improved orientation estimation method based on *Extended Kalman Filter* (EKF) with gyroscope. And our main goal here is to optimize the process and get more precise results using data fusion from multiple sensors. Our basic idea is first to get an initial orientation estimation based on accelerometer and magnetometer and then build an estimation model through gyroscope and calibrate through EKF.

Note that the orientation of smartphone could be considered as the Euler Angle rotated around $X$, $Y$, and $Z$, the three axes of body frame coordinate sequently. The angle is presented as: *pitch* $(\theta)$, *roll* $(\phi)$, and *yaw* $(\varphi)$, respectively. The orientation could be computed by

$$\theta = \arctan\frac{G_u x}{G_u x}, \quad \phi = \arcsin G_u y, \quad \varphi = \arctan\frac{E_u y}{N_u y}, \tag{1}$$

where $G_u$ is the unit vector of the acceleration and both $E_u$ and $N_u$ are the unit vector of y direction and the magnetic north, respectively.

After we fixed the smartphone on the windshield, we first wait for approximately 10 s to calculate the initial orientation. The reason for this process is to eliminate the device-dependent errors and fluctuation on sensory data. We take the average gravity, and magnetic field as $\overline{G_1}$, and $\overline{B}_1$, respectively, and calculate the initial orientation based on Eq. (1).

In the following sensor fusion process, the *estimated orientation* and *measured orientation in EKF* are based on gyroscope and a combination of accelerometer and magnetometer. The orientation in time $k$ is related to the posteriori estimation of orientation in time $k$−1. We present the non-linear stochastic difference equation as:

$$q_k = \widetilde{q}_k + A\left(Q_{am} - h_k\widetilde{q}_k\right), \widetilde{q}_k = R_k + q_{k-1}, \tag{2}$$

where $q_k$ indicates the estimated orientation at time $k$, $Q_{am}$ denotes the orientation based on current accelerometer and magnetometer, and $\widetilde{q}_k$ presents the priori estimate at time $k$, which is calculated through last orientation and rotation increment. Here, $A$, $h_k$, and $R_k$ are the Kalman gain, measurement matrix, and rotation increment, respectively.

The rotation increment is the changing rotation angle at time slot $k$, and $\overline{R_k}$ is given as:

$$\overline{R}_k = \Omega \cdot \left[pitch\ roll\ yaw\right]^T \cdot \Delta t, \tag{3}$$

where $\Delta t$ is the time interval of sampling, and $\Omega$ is

$$\Omega = \begin{bmatrix} 1 & \sin\phi \cdot \tan\theta & \cos\phi \cdot \tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\tan\theta & \cos\phi/\tan\theta \end{bmatrix}. \tag{4}$$

The EKF will bring a continuous changing orientation of smartphone, and the improved orientation in the following time slots will be:

$$\widetilde{q}_k = q_{k-1} + \overline{R}_k. \tag{5}$$

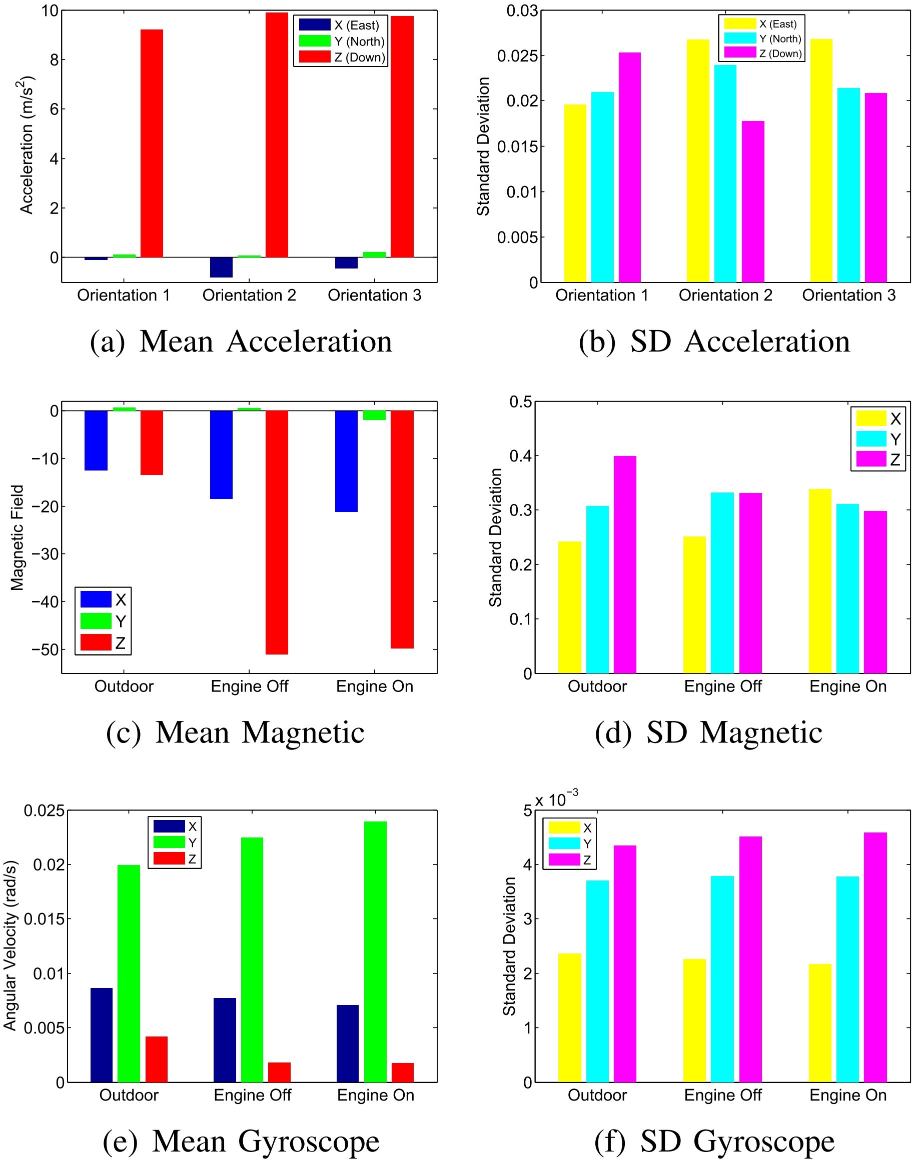


**Fig. 3** Mean values and standard deviations for accelerometer, magnetometer, and gyroscope respectively for different cases

Rotating the linear acceleration from BFC to EFC, the calculating of velocity and driving distance from the starting point is feasible.

### 5.3 Noises in distance estimation

In SmartLoc, the system updates the latest location for a certain time interval, denoted as $\Delta t$; therefore, the distance from starting point after one time slot could be described by the linear state equation:

$$\widetilde{X}_k = X_{k-1} + V_{k-1} \cdot \Delta t + \frac{1}{2} \cdot a_k \cdot \Delta t^2, \tag{6}$$

where $\widetilde{X}_k$ is the current estimated distance from the starting point at time slot $k$, $X_{k-1}$ is the distance from starting point at previous time slot, $V_{k-1}$ is the velocity of vehicle at beginning of time slot $k$, and $a_k$ is the acceleration during time slot $k$. Equation (6) could be presented recursively as:

$$\widetilde{X}_k = \sum_{i=0}^{k-1} V_i \cdot \Delta t + \frac{1}{2} \cdot \Delta t^2 \cdot \sum_{i=0}^{k} a_i. \tag{7}$$

However, in practice, both $V_i$ and $a_i$ contain large noises. Suppose the noises of $V_i$ and $a_i$ obey normal distributions, and the standard deviations of these noises are

$\sigma_1$ and $\sigma_2$, respectively. The standard deviation on the estimated distance error will be: $(k-1)\cdot\sigma_1\cdot\Delta t+\frac{1}{2}\cdot\Delta t^2\cdot k\cdot\sigma_2$. Therefore, with the increase of dead reckoning duration ($k\Delta t$), the accumulated errors will increase more than linearly. Such errors, in a long run, are inevitable, and our extensive real world experiments confirm it.

## 6 Advanced calibration

In SmartLoc, both GPS and inertial sensors could provide a possible current location while driving. The main idea of SmartLoc is to use the dead reckoning from sensing data to compensate the GPS location when GPS signal is weak and to use the GPS signals to build a better dead reckoning model from the sensing data when the GPS signal is strong. Notice that we cannot neglect the fact that GPS signals still contain errors even if the environment is ideal. In addition, while driving in a crowd city, especially in big metropolises such as Chicago, complicated road infrastructures may cause large GPS errors. However, in SmartLoc, such infrastructures may offer unique information to improve the location accuracy. In this section, we present our advanced calibration techniques in GPS positioning and then the correction by road infrastructures.

### 6.1 GPS compensation

We then establish a model to calibrate the location based on both GPS information and sensory data.

**Speed estimator:** The true velocity $V_i$ at the end of a timeslot $i$ could be represented as

$$V_i = V_{i-1} + \beta \cdot a_s \cdot \Delta t + \mu, \tag{8}$$

where $\beta$ is the parameters learnt and adjusted in real time, $a_s$ is the mean of the measured acceleration data during the timeslot, and $\mu$ is the noise. Thus, when GPS signal is strong, we will compute $V_i$ and $V_{i-1}$ from GPS values. The measured acceleration $a_s$ will be computed from accelerometer. Then, using linear regression, we find the best $\beta$.

When GPS signal is weak, we will use the model

$$V_i \leftarrow V_{i-1} + \beta \cdot a_s \cdot \Delta t$$

to estimate the velocity $V_i$.

**Distance estimator:** When GPS signal is strong, let $G(\Delta(t))$ be the moving distance during timeslot $i$ from GPS, which still contains errors (denoted as $\eta$). This distance provided by GPS could be denoted as $G(\Delta t) = \lambda_1\cdot V_{i-1}\cdot\Delta t+\frac{1}{2}\cdot a_r\cdot\Delta t^2+\eta$, where $a_r$ is the actual unknown acceleration in timeslot $\Delta t$. Here, $\lambda_1$ is used to reflect possible error in the estimated speed $V_{i-1}$ for timeslot $i-1$. We only know the measured acceleration $a_s$, which could be denoted as $a_s = (1+\varepsilon)a_r + \delta$. In this work, we use the following formula to estimate the distance $G(\Delta t)$:

$$G(\Delta t) = \lambda_1\cdot V_{i-1}\cdot\Delta t+\lambda_2\frac{1}{2}\cdot a_s\cdot\Delta t^2+\lambda_3\cdot\Delta t^2+\lambda_4\cdot\Delta t+\eta, \tag{9}$$

where $\lambda_i$ are the parameters to be learnt by linear regression. When GPS signal is strong (GPS error is $\leq$20 m), based on the value of $V_{i-1}$ and $a_s$ estimated from the sensor data in every sample point, we train the parameters $\lambda_i$ in Eq. (9). These parameters will be used by dead reckoning to predict the moving distance $S(\Delta t)$ in the timeslot $i$ when GPS signal is as weak as the following:

$$S(\Delta t) \leftarrow \lambda_1\cdot V_{i-1}\cdot\Delta t+\lambda_2\frac{1}{2}\cdot a_s\cdot\Delta t^2+\lambda_3\cdot\Delta t^2+\lambda_4\cdot\Delta t. \tag{10}$$

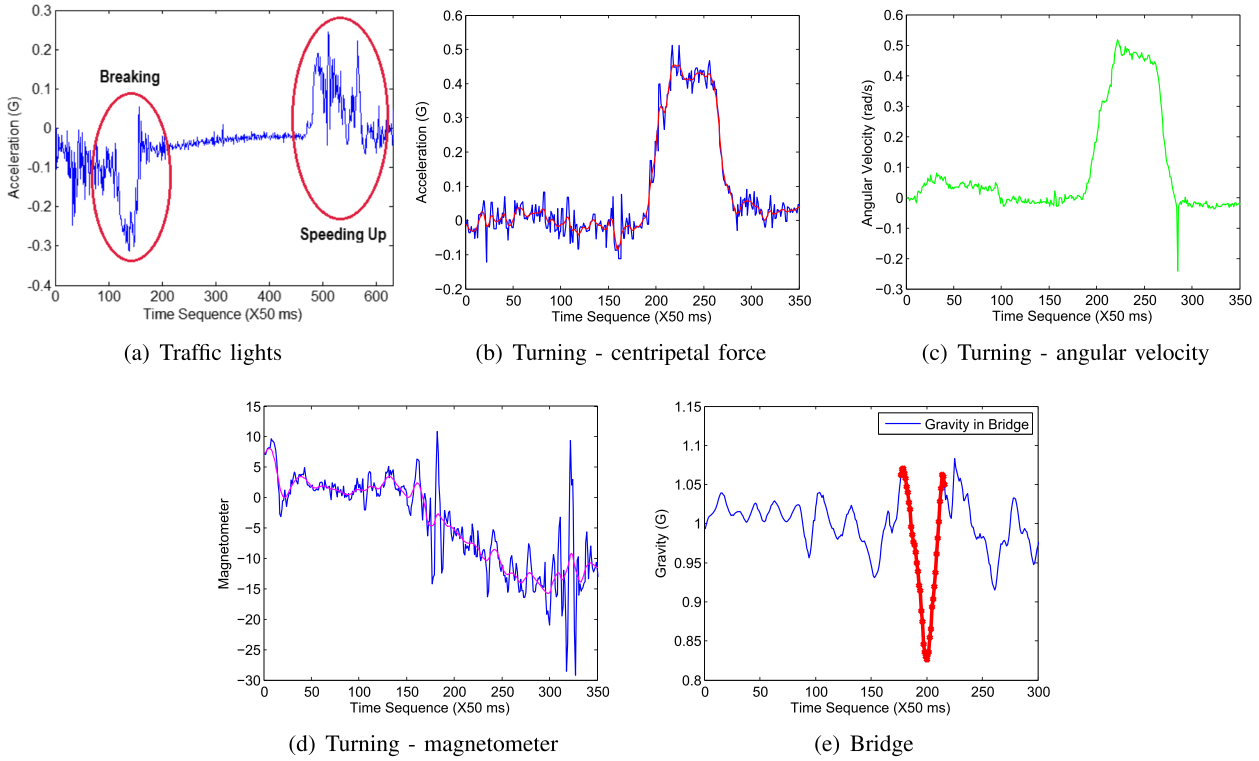
From the estimated moving distance $S(\Delta t)$, we can estimate the location at timeslot $i$.

### 6.2 Identify driving condition and landmarks

As we mentioned previously, the road infrastructures, including tunnels, bridges, crossroads and traffic lights, cause large noises in the data collected from inertia sensors, which also result in a large drift in the estimated moving distance if they are not treated rigorously. In this work, we exploit the precise location of these infrastructures that are available in Google Map to calibrate the localization as landmarks without any extra cost.

**Traffic light:** While driving in many large well-organized cities, such as Chicago, one often encounters traffic lights. When the vehicle stops by the traffic lights, and then drives through the crossroad, there is a unique pattern reflected by motion sensors, as shown in Fig. 4a. When vehicles face traffic lights, the pattern could be considered as the combination of braking and speeding up. The acceleration falls below zero when the car brakes, reaching the lowest point at the very moment when the vehicle stops, and gets back to zero swiftly. As soon as the green light turns on, the vehicle speeds up with the acceleration increases. However, in rush hours, the real location where the car stops may not be at the crossroad but with a certain distance away from the crossroad. In this case, SmartLoc will adjust the moving distance based on the estimated stopping location from empirical data, i.e., subtracting the distance from the car to the crossroad.

**Turning:** Sometimes, vehicles may face both traffic lights and turning at the same time. During the turning, all the motion sensors will detect the turning. Figure 4b indicates the centripetal force sensed by the accelerometer, and the scale of the acceleration depends on the speed that the vehicle is turning. Simultaneously, the



(a) Traffic lights (b) Turning - centripetal force (c) Turning - angular velocity

(d) Turning - magnetometer (e) Bridge

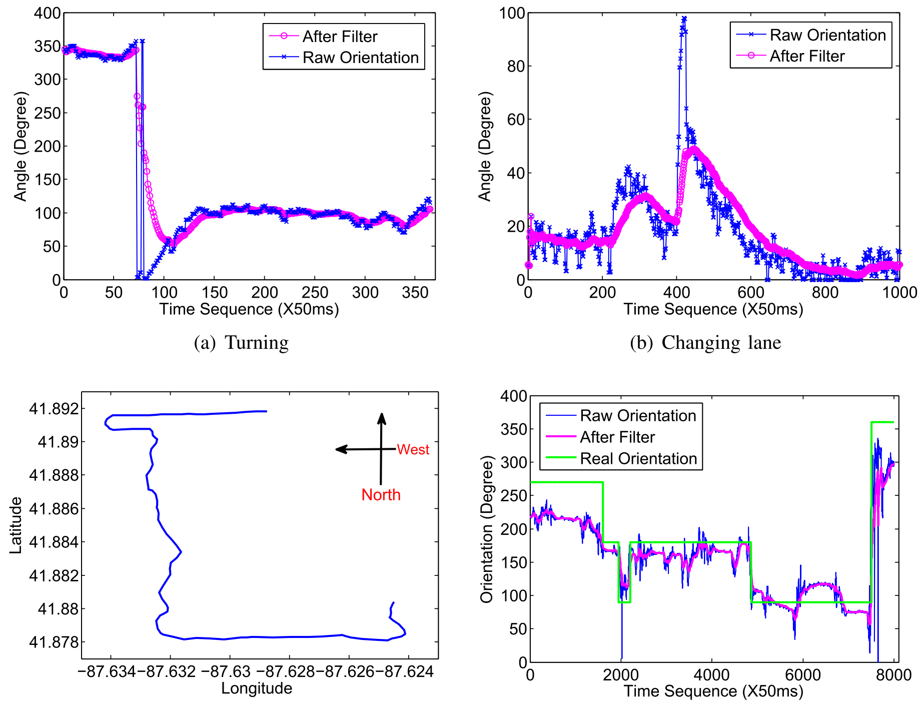
**Fig. 4** Patterns of the sensor data collected in different road infrastructures when driving. **a** Car stopping and crossing a traffic light. **b**, **c**, **d** Car turning 90$^o$. **e** Car crossing a bridge

angular velocity sensed by gyroscope also reaches up to 0.5 $rad/s$ in our test case in Fig. 4c, and the data from the magnetometer changes as well, with large fluctuations as shown in Fig. 4d. Finally, the orientation of smartphone will also change approximately 90$^o$ when turning left or right. Such angle change is observed mainly along the $Z$ axis on EFC, and the readings of 0, 90, 180, 270 represent north, east, south, and west, respectively. Although the angle may not be accurate enough due to the large noises in magnetometer (the maximum error we experienced in our tests was about 30$^o$), we can still correctly determine the road segment to which the car is turning. In Fig. 5a, the vehicle turns from north, the angle changes from about 350$^o$ to 100$^o$, which is east. We also compare the measured angle difference for turning and changing lanes, as shown in Fig. 5b. The amplitude of angle difference when a car changes lanes is much smaller than that when a car turns into a different road segment. In addition, we calculate the standard deviation for angle differences for lane changing, which is below 10 in most of the cases. Thus, determining the turning and the direction of driving is feasible. We conduct more study on the driving orientation estimation. Figure 5c plots the raw trace of the vehicle through GPS, where the GPS signal is good. And Fig. 5d illustrates the corresponding raw orientation generated through inertial sensors. We employ slide window to eliminate some noise and calculate the driving orientation, which matches the ground truth.

**Bridges:** Other possible road infrastructures that a vehicle may experience are bridges and tunnels. In our measurement, such patterns are more obvious and easier to be detected. Having been converted from BFC to EFC, the acceleration along the gravity will be less than 1$G$ when the car is driving uphill and larger than 1$G$ when downhill. A bridge or a tunnel often has both uphill and downhill pattern, as shown in Fig. 4e.

## 7 Experiments and evaluations

We conduct extensive evaluations of SmartLoc in different scenarios, both in downtown Chicago and on nearby highways. In our evaluations, a Samsung Galaxy S3 with latest Android 4.1.2 is mounted to the windshield. We drive for over 100 different road segments in downtown Chicago ranging from 1 to 10 km and over 30 km on nearby highways. Since the motion sensor will provide the orientation of driving, knowing the driving distance from the starting point and the driving orientation, one can compute



(a) Turning

(b) Changing lane

(c) Driving GPS trace

(d) Estimated orientation

**Fig. 5** Readings from turning and changing lanes

the real-time location easily. Therefore, the key problem becomes estimating the driving distance. The evaluation metrics include the traveling distance, road infrastructure recognition, accuracy, and energy consumption.

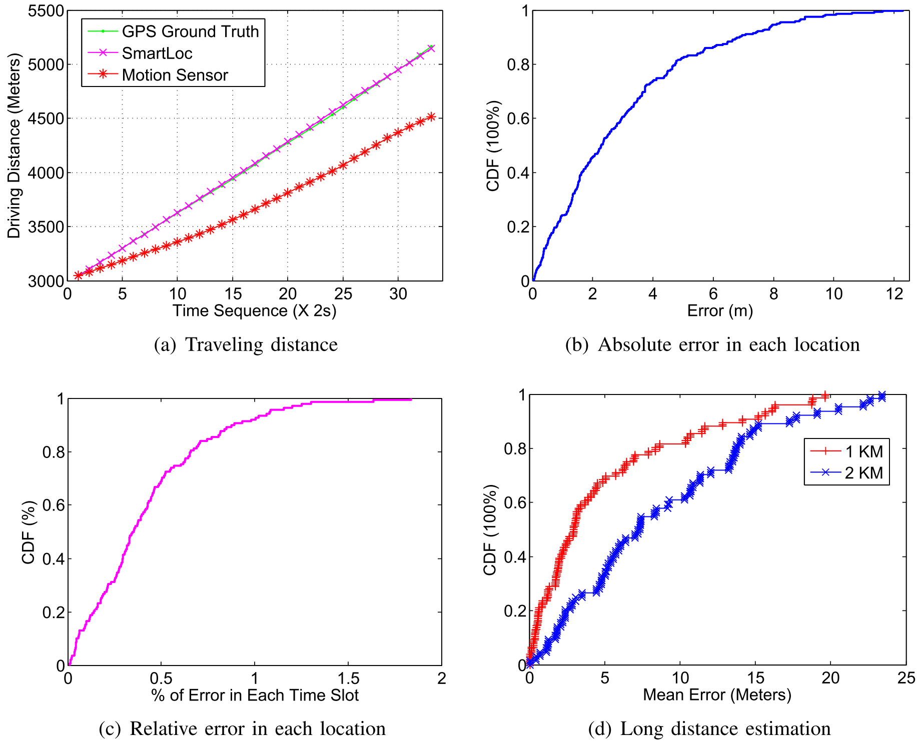
## 7.1 Traveling distance estimation

First of all, the typical frequency of updating position through GPS in a smartphone is much lower than that of motion sensors: it takes around 2 s on average to update GPS location. We denote the driving distance in a time slot as a *traveling segment*. We focus on the evaluation of traveling distance estimation in two aspects: (1) the accuracy in distance estimating in traveling segments and (2) the accuracy in final distance estimation of long road segments.

### *7.1.1 Evaluation on highway*

We first evaluate SmartLoc when driving on the highway, where the driving velocity is relatively stable and there are no stops most of the time. We drive over ten different highway segments, where total distance is over 60 km (with driving speed in the range of about 100 km/hr to about 120 km/hr). The smartphone has access to the precise location information from the GPS, which is updated every 2 s and could be considered as the ground truth. Meanwhile, we collect the readings from the motion sensor and calculate the parameters in linear regression for 3 km. Then, we predict the traveling distance for the next 2 km and compare the distance among ground truth, SmartLoc, and pure motion sensor.

Figure 6a illustrates the comparisons of driving distance estimation using SmartLoc (with motion sensors) and GPS. The ground truth (GPS readings) is plotted by the green curve. It is obvious that the errors between the distance from motion sensors and the ground truth are becoming larger along the time. The reason is that the errors will be accumulated without any calibration. Through learning techniques, SmartLoc calculates suitable parameters and apply them into the prediction, so that the estimation errors are much smaller. Figure 6b indicates that the largest error is about 12 m among the ten different highway segments (each of length 2 km), and over 80 % errors are less than 5 m. Compared to the actual distance extracted from the ground truth, as shown in Fig. 6c, over 95 % errors (among all locations where GPS location can be extracted) are less than 1 % of the actual driving distance, and the largest error is less than 2 % of the actual driving distance. We also notice that the accuracy of predicting the traveling distance will decrease with the increase of driving distance. We predict the driving distance for both 1 and 2 km after taking the data of the first 3 km to build a dead reckoning model. In our experiments, 80 % of the errors in the first group of the cases (predicting 1 km) are less than 10 m, but they become



(a) Traveling distance

(b) Absolute error in each location

(c) Relative error in each location

(d) Long distance estimation

**Fig. 6** Traveling on a highway

15 m in the cases of predicting 2 km. The largest error is around 19.8 and 23 m, respectively. We plot the results in CDF as shown in Fig. 6d.

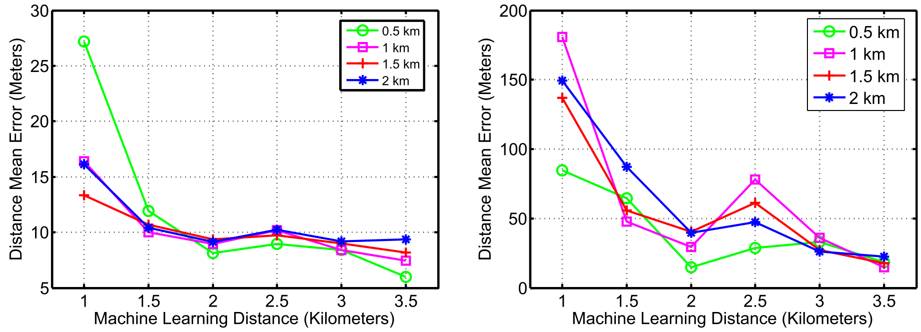
However, based on the evaluation, we discover that the estimation results cannot maintain high accuracy for a long distance. The main reasons come from the user dependent driving behavior and unpredictable road conditions. We find that SmartLoc has better estimation accuracy when the driving speed remains stable. When the driving speed fluctuates frequently, the predicted results by SmartLoc divert from the actual value. Calibrating the location periodically is a feasible way to improve the location accuracy in real applications.

#### *7.1.2 Evaluation in city without using landmarks*

We then test SmartLoc in a city scenario but without using landmarks. We drive in downtown Chicago for over 30 different roads, where some roads have reliable GPS signals and some do not. We separate these roads into road segments (more than 100 road segments are produced) whose sizes are determined by our evaluations presented in the rest of this section. Similarly, we analyze the performance of SmartLoc in two phases as aforementioned.

Note that GPS signals in downtown Chicago are relatively poor and also time dependent. It is difficult to get the ground truth of all locations through smartphones. To address this, we evaluate SmartLoc in the peripheral area of downtown Chicago, where the buildings are not so high as in the center of downtown, and GPS signal is relatively good. It is much easier to get the ground truth through GPS. In our evaluations, we assume that some of the road segments have poor GPS (by removing the GPS values) so that we can apply SmartLoc to calculate the location in these areas and compare our results with the ground truth.

Obviously, the driving habit and road conditions in a city are different from those on the highway, and slight decelerating will lead to the deviation between the estimated results and the ground truth. We first evaluate the reliability of the SmartLoc system when different driving distances are used to train the system, ranging from 0.5 to 3.5 km. Generally speaking, the accuracy will increase when the learning distance increases, as illustrated in Fig. 7a. In this figure, the $X$ axis indicates the driving distance used for training our dead-reckoning model in SmartLoc, and the $Y$ axis represents the mean distance (between the actual location reported by GPS and the location estimated by SmartLoc) in every time slot when we sample GPS locations (i.e., every 2 s, or about every 22 m when driving at speed about 40 km/hr). This experiment measures the accuracy for the prediction when we drive for over four different road segments with length from 0.5 to 2 km (24 different cases in total). Due to the unstable driving behavior, short road segments for



(a) Mean error in each slot

(b) Mean overall distance error

**Fig. 7** Accuracy vs. learning distance

training of SmartLoc may lead to a large estimation error in each time slot. When SmartLoc only learns with the trace of 1 km, the mean error in every time slot in different scenarios is around 15 m (the largest one is nearly 30 m). When SmartLoc trains the dead-reckoning model using longer trace, the mean estimation error decreases in all testing cases. The smallest error is less than 6 m, which is less than half of the error when the training trace is 1 km. We also observe that the error grows with the increase of the length of the testing road segment in most scenarios. For example, by training SmartLoc with a trace of 3.5 km, the mean error of estimating all locations in a 2-km road segment is nearly twice of that when testing a 0.5-km road segment.

We then evaluate the error on estimating the overall driving distance, as shown in Fig. 7b. We test SmartLoc for all the road segments and measure the error between the predicted driving distance and the actual driving distance for each segment (of all segments with distance from 0.5 to 2 km) under different training traces. If SmartLoc only learns with 1 km trace, the parameters in Eq. (9) cannot be computed accurately enough. Thus, the estimation error increases to 180 m in all our tests. When SmartLoc learns enough samples, the parameters are much more reliable, and the average accumulated error is below 30 m, which is significantly better than the GPS in Chicago downtown.

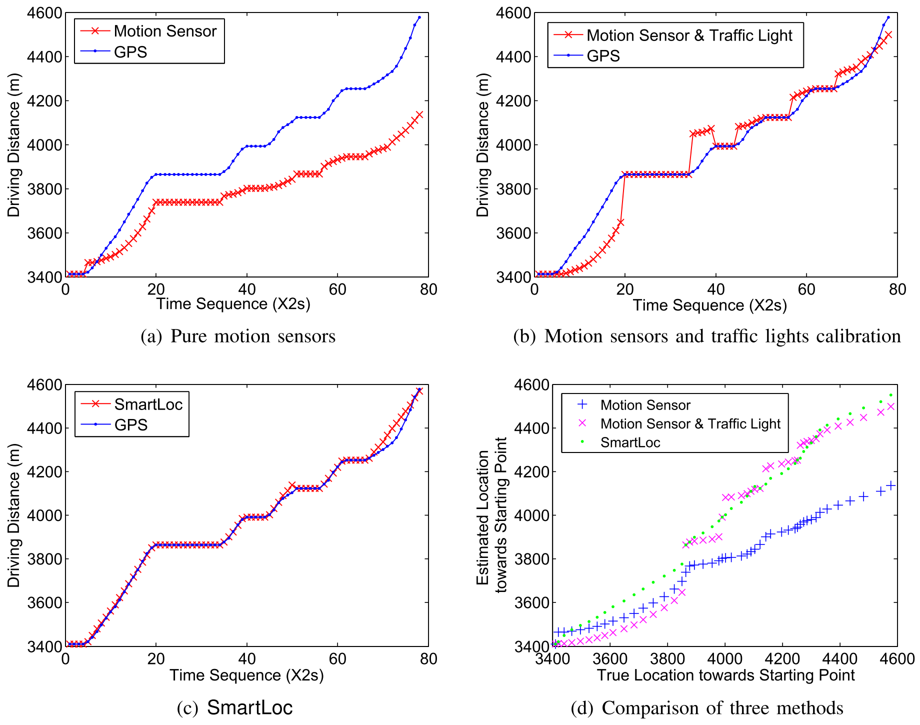
### *7.1.3 Evaluation in city with landmarks*

SmartLoc could calibrate the location as soon as it detects specific patterns for landmarks, especially traffic lights and turnings. Since, in the city, the probability of facing green lights in all the crossroads or always driving on a straight street is extremely low, we can take advantage of these driving features and landmarks to improve the estimation accuracy so that we do not have to estimate for a long distance. We then extensively evaluate the performance of SmartLoc with the calibration using such landmarks.

We compare the performances of three different methods: using SmartLoc without noise canceling and landmark calibration (called *motion sensor*), using SmartLoc calibration by traffic lights (called *motion sensor and traffic light*), and SmartLoc using calibration from all detected landmarks. In this experiment, we assume the first 3400 m are with reliable GPS signals and the precise locations are accessible. The estimation starts from 3400 m, and the first three subfigures in Fig. 8 indicate the driving distance from the starting point versus time dimension.

In Fig. 8a, for the method using mobile sensors but without any calibration or noise canceling, the double integration on acceleration leads to the final deviation of over 400 m after driving about 1200 m. When the road pattern detection is introduced, the location will be calibrated when SmartLoc senses the road infrastructure patterns. During the same experiment, our vehicle crosses five traffic lights in total, and successfully detected all five traffic lights. The estimated locations are all then adjusted accordingly. From Fig. 8b, the errors are still high, especially in the crossroads. Surprisingly, after integrating the machine learning and noise canceling, SmartLoc's result is much better, as shown in Fig. 8c. For about the first 900 m, the curve of SmartLoc nearly overlaps with the curve of the ground truth. For the first 450 m, the vehicle passes three crossroads with all green lights, and the error is less than 20 m most of the time. After the final traffic light, the vehicle has to drive in a relatively low speed because of the road construction. The predicted distance, consequently, deviates from the ground truth slightly, but at the ending of the road, the errors remain small. We plot all the estimated distances by three methods in Fig. 8d, with the $X$ axis being the ground truth distance and the $Y$ axis being the predicted distance. SmartLoc results are distributed around the diagonal, and pure motion sensor approach deviates greatly.

The deviation of the results from the ground truth comes from the accumulated errors from all time slots. For the previous experiments, we also plot the errors



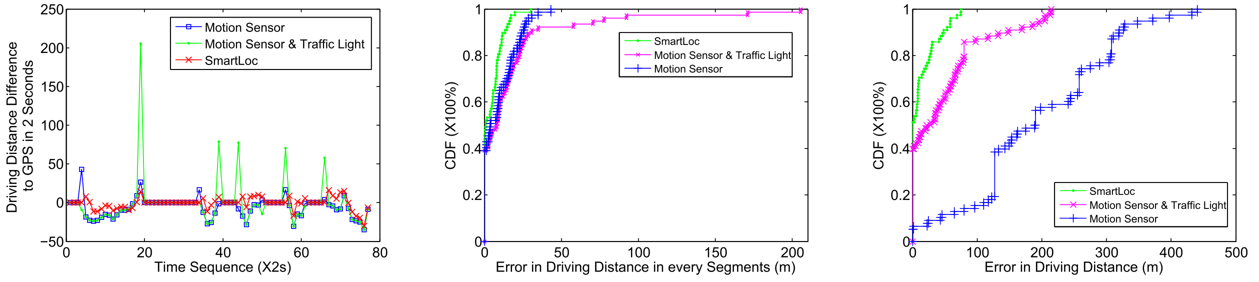
(a) Pure motion sensors (b) Motion sensors and traffic lights calibration (c) SmartLoc (d) Comparison of three methods

**Fig. 8** Distance prediction comparison among three methods and ground truth

in every time slot in Fig. 9a. SmartLoc system using all detected landmarks for the calibration has the smallest mean error of estimated location for all time slots: 90 % of them are lower than 20 m from the CDF in Fig. 9b. The other two approaches have larger errors. Figure 9c describes the CDF of the total driving distance errors.

#### 7.1.4 *Evaluation analysis*

Based on the evaluation results presented in this section, an obvious conclusion is that SmartLoc could provide precise driving distance estimation in certain scenarios. In every time slot, the driving distance is estimated through current sensory data and parameters from linear regression. Suppose the error (denoted as $D_i$) in the estimation of each time slot $i$ follows normal distribution: $D_i \sim N\left(\mu, \sigma^2\right)$, with mean $\mu$ and variance $\sigma^2$. Then, the estimation of total traveling distance $S_t$ in a period of $t$ timeslots is bounded by the summation of traveling distance in all time slots: $S_t = \sum_{i=1}^{t} D_i$. In this case, the error, from a long-term perspective, will be accumulated. Obviously, $S_t \sim N(t\mu, t\sigma^2)$. The variance of the variable $S_t$ will be $t\sigma^2$. Thus, the mean error will often increase along the time, which leads to the conclusion that it is difficult to predict



(a) Driving distance error in every slot (b) CDF of distance error in every slot (c) CDF of error in long distance esti.

**Fig. 9** Comparison of three methods on driving distance errors

the traveling distance precisely in the long term, although sometimes the deviation in some continuous timeslots may be neutralized. For a given error bound $\delta$, $Pr(S_t \geq \delta)$ is higher when $t$ is larger.

## 7.2 Localization in the city

We then present the localization results of SmartLoc in Chicago downtown. As aforementioned, it is difficult to get the ground truth for the majority of the sampling locations. We set the goal of localization is to find the final location. Since, Section 3 has demonstrated that there are nine bad road segments with lengths over 400 m, which is less than three blocks in downtown Chicago. The goal of SmartLoc is then to obtain a relatively accurate location estimation within three blocks. We randomly select 100 points as destinations in the experiment, and a destination could be one block, two blocks, or three blocks away from the starting point. We drive to these destination points to evaluate if the location of the destination can be precisely calculated by SmartLoc. We assume that the GPS signals are good before the starting point, and SmartLoc trains the dead-reckoning model during the driving. In this experiment, we test the accuracy of estimating the location of each sampling point in every time slot and of estimating the location of the final destination, as shown in Fig. 10a, b, respectively. When SmartLoc only navigates to the destination within one block, with probability 70 %, the error of estimating the location for each sampling slot is less than 10 m, and with probability 85 %, the mean error is less than 30 m. When the destination is two blocks away, about 75 % of the errors are less than 30 m; when the destination is three blocks away, about 80 % errors are less than 50 m. From these results, the error of destination locating within a few blocks is acceptable. We also plot the localization results for one road segment with length over 6400 m in Fig. 11. In this figure, the red spots denote the ground truth generated from GPS, and the blue spots represent the localization calculated by SmartLoc, where the green line between them is the localization error for every location. It is clear that with longer segment the accuracy of localization is pretty good.

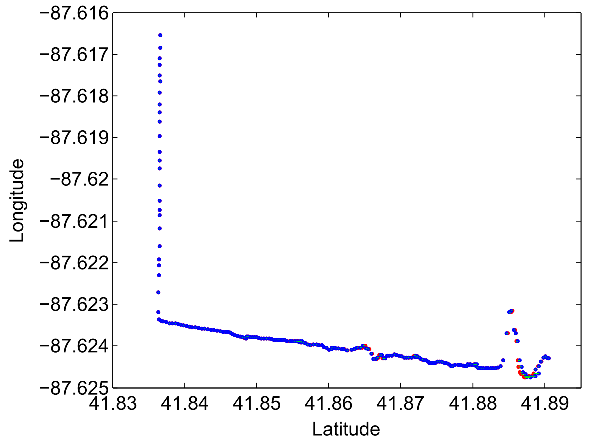
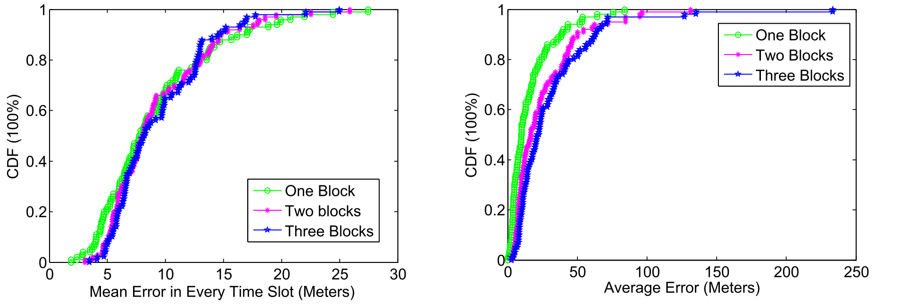


**Fig. 11** Localization on a single long street

## 7.3 Energy consumption analysis

SmartLoc provides the location through both GPS and inertial sensors. Empirically, the profiles of energy consumption for GPS module and inertial sensors in smartphones are different. The average electric current of GPS module in Android smartphone is approximately 135 mA, which is larger than that of motion sensors [54]. In addition, Android API provides four rate modes of delivering sensory data, which are Fastest, Game, UI and Normal. The Fastest mode delivers the sensory data without any delay; thus, the energy consumption is the largest (95 mA [54]), while the Normal is the slowest, and also the most energy efficient one (15 mA [54]). Clearly, SmartLoc can reduce the energy consumption by leveraging inertial sensors instead of GPS compared with the approach using only GPS.

Suppose the energy cost of a GPS module for every time slot is $E_{GPS}$, and the energy cost of inertial sensors are $E_{AMG}$. Obviously, $E_{AMG} \leqslant E_{GPS}$. Generally, the main



(a) CDF of error of location estimation in every sampling timeslot (b) CDF of error in locating the final destination in different blocks

**Fig. 10** Localization performance evaluation of SmartLoc in city

purpose of the GPS is calibration and training, as shown in Fig. 12, where *GPS* indicates the localization through the GPS, and *AMG* denotes the location estimation through motion sensors. Initially, during $T_1$, the system will train the model and calculate suitable parameters through the linear regression, and such parameters will be employed in estimating the trajectory in $T_2$. Periodically, SmartLoc will calibrate the location from dead-reckoning when GPS signal becomes good again in $T_3$, and the parameters will be updated. Then the ratio of saved energy will be $\frac{(E_{GPS}-E_{AMG})\cdot T_2}{E_{GPS}\cdot(T_2+T_3)}$. The durations of $T_1$, $T_2$, and $T_3$ will affect the acceptable accuracy and energy consumption of localization of SmartLoc. Our first experiment is taken under the circumstance of activating all the motion sensors and GPS modules, the driving duration lasts for 30 min, and the battery of smartphone discharges from 84 to 70 %. Similarly, GPS localization for half an hour will lead to the drop of battery from 70 to 61 %. In most of our cases, in order to get higher accuracy, the ratio between training duration and estimating duration is ranging from 6:3 to 8:3. For example, in the experiment shown in Fig. 8c, the training distance is about 3400 m and SmartLoc estimates the following 1200 m (with GPS off), which reduces the energy by approximately 10 % without affecting the localization accuracy. Note that as long as the GPS signal is strong during $T_1$ and $T_3$, we can turn off GPS in $T_2$. If GPS signal is weak in $T_1$ or $T_3$, we can keep the GPS alive for longer periods. Another possible working pattern is turning the GPS on/off according to the condition of the GPS signal. When GPS signal is good, SmartLoc turns the GPS to calibrate the system parameters, turns off the GPS when signal is weak, and then use the newly updated parameters to calculate the driving speed and distance.

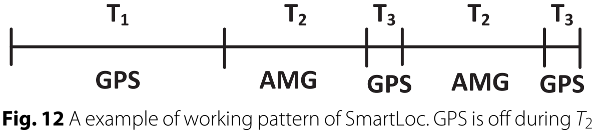


**Fig. 12** A example of working pattern of SmartLoc. GPS is off during $T_2$

## 8 Conclusions

This paper presented the design and evaluation of SmartLoc, an urban localization system through inertial sensors in smartphones, which can be used for many location-based smart city and CPS applications. We showed that pure dead-reckoning cannot provide acceptable localization accuracy. SmartLoc detects the road infrastructures and driving patterns as landmarks to calibrate the localization results. We establish a model to estimate the trajectory using linear regression method. Our extensive evaluations indicate that SmartLoc improves the location accuracy to be acceptable for about 85 % of the cases in Chicago downtown, compared with about only 75 % using raw GPS data.

This is only the first step-stone for localization in metropolises. Additional functionality and techniques could further improve the utility of SmartLoc. For instance, for long road segments without good GPS signal, additional data, such as GSM signal and driving history, could be used to improve the accuracy. WiFi data collected on the road-side may offer new side-information for improving the accuracy as well. Although the WiFi-based approach methods suffer signal fading and multi-path effect in an indoor environment, the signal propagation may follow more regular pattern in an outdoor environment. A cloud-based social network may provide drivers more road information, which may further improve the accuracy on unpredictable roads. Last, by adjusting the driving behavior and road information in different cities, SmartLoc can have potential applications in wider regions. These directions will be left as our future works. However, we strongly believe that the accurate urban localization is critical for smart city and CPS applications and can be achieved by smartly utilizing mobile data from both inertial sensors and GPS.

## Endnote

[1] The accelerometer values are the acceleration speeds in three axes of the BFC (coordinate is uniquely determined by the orientation of the smartphone). On the other hand, for localization, we need to compute the integration of acceleration in EFC. It has been well-documented in the literature (e.g., [13]) that the rotation of smartphones will introduce large errors in speed and location estimation.

### Competing interests

The authors declare that they have no competing interests.

**Acknowledgements**

The work is partially supported by the US National Science Foundation under Grant No. NSF CNS-1319915, CNS-1343355, CNS-1526638, EECS-1247944, and CMMI-1436786, and the National Natural Science Foundation of China under Grant No. 61272426, 61379117, 61428203, 61520106007, and 61572347.

**Author details**

[1]Department of Computer Science, University of North Carolina at Charlotte, 9201 University City Blvd., Charlotte, NC 28223, USA. [2]Department of Computer Science, Illinois Institute of Technology, 10 West 31st Street, Chicago, IL 60616, USA. [3]School of Software, Tsinghua University, 30 Shuangqing Rd, Haidian, Beijing 100084, China. [4]School of Computer Science and Technology, University of Science and Technology of China, Huang Shan Road, Hefei, Anhui 230027, China. [5]College of Information Engineering, Taiyuan University of Technology, 79 Yingze W Street, Taiyuan, Shanxi 030024, China.

Received: 22 December 2015 Accepted: 4 April 2016
Published online: 19 April 2016

**References**

1. Y Cheng, Y Chawathe, A LaMarca, J Krumm, in *Proc. of ACM MobiSys*. Accuracy characterization for metropolitan-scale Wi-Fi localization (ACM, New York, USA, 2005), pp. 233–245

2. S Syed, E Cannon, Map-aided GPS navigation: linking vehicles and maps to support location-based services. GPS World. **16**(11), 39–44 (2005)
3. W-W Kao, in *Vehicle Navigation and Information Systems Conference, 1991*. Integration of GPS and dead-reckoning navigation systems, vol. 2 (IEEE, Piscataway, New Jersey, USA, 1991), pp. 635–643
4. P Zhang, C Sadler, S Lyon, M Martonosi, in *Proc. of ACM SenSys*. Hardware design experiences in zebranet (ACM, New York, USA, 2004), pp. 227–238
5. B Thorstensen, T Syversen, T Bjørnvold, T Walseth, in *Proc. of ACM MobiSys*. Electronic shepherd-a low-cost, low-bandwidth, wireless network system (ACM, New York, USA, 2004), pp. 245–255
6. L Mo, Y He, Y Liu, J Zhao, S Tang, X Li, G Dai, in *Proc. of ACM SenSys*. Canopy closure estimates with Greenorbs: sustainable sensing in the forest (ACM, New York, USA, 2009), pp. 99–112
7. H Wang, T Zhu, S Lei, W Yang, Y Wang, CMTG: A content-based mobile tendency geocast routing protocol in urban vehicular networks. Int. J. Distributed Sensor Netw. **2015** (2015). Article ID 163157
8. F Li, L Zhao, X Fan, Y Wang, Hybrid position-based and DTN forwarding for vehicular sensor networks. Int. J. Distributed Sensor Netw. **2012** (2012). Article ID: 186146, 10
9. Y Wang, Y Ge, W Wang, W Fan, in *Proceedings of 2015 Workshop for Big Data Analytics in CPS: Enabling the Move From IoT to Real-Time Control*. Mobile big data meets cyber-physical system: Mobile crowdsensing based cyber-physical system for smart urban traffic control, (Seattle, Washington, 2015)
10. B Guo, Z Wang, Z Yu, Y Wang, N Yen, R Huang, X Zhou, Mobile crowd sensing and computing: the review of an emerging human-powered sensing paradigm. ACM Comput. Surv. **48**(1) (2015). Article 7
11. L Ojeda, J Borenstein, in *Proc. of International Society for Optics and Photonics Defense and Security Symposium*. Non-GPS navigation with the personal dead-reckoning system, (2007). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.74.4667&rep=rep1&type=pdf
12. A Varshavsky, M Chen, E de Lara, J Froehlich, D Haehnel, J Hightower, A LaMarca, F Potter, T Sohn, K Tang, et al., in *Proc. of IEEE WMCSA*. Are GSM phones the solution for localization? (IEEE, Washington, DC, USA, 2006), pp. 34–42
13. S Guha, K Plarre, D Lissner, S Mitra, B Krishna, P Dutta, S Kumar, in *Proc. of ACM SenSys*. AutoWitness: locating and tracking stolen property while tolerating GPS and radio outages (ACM, New York, USA, 2010), pp. 29–42
14. J Liu, B Priyantha, T Hart, H Ramos, A Loureiro, Q Wang, in *Proc. of ACM SenSys*. Energy efficient GPS sensing with cloud offloading (ACM, New York, USA, 2012)
15. K Lin, A Kansal, D Lymberopoulos, F Zhao, in *Proc. of ACM MobiSys*. Energy-accuracy trade-off for continuous mobile device location (ACM, New York, USA, 2010), pp. 285–298
16. H Hassanieh, F Adib, D Katabi, P Indyk, in *Proc. of ACM MobiCom*. Faster GPS via the sparse Fourier transform (ACM, New York, USA, 2012), pp. 353–364
17. I Constandache, R Choudhury, I Rhee, in *Proc. of IEEE INFOCOM*. Towards mobile phone localization without war-driving (IEEE, Piscataway, New Jersey, USA, 2010), pp. 1–9
18. H Wang, S Sen, A Elgohary, M Farid, M Youssef, R Choudhury, in *Proc. of ACM MobiSys*. No need to war-drive: unsupervised indoor localization (ACM, New York, USA, 2012), pp. 197–210
19. Z Yang, C Wu, Y Liu, in *Proc. of ACM MobiCom*. Locating in fingerprint space: wireless indoor localization with little human intervention (ACM, New York, USA, 2012), pp. 269–280
20. H Liu, Y Gan, J Yang, S Sidhom, Y Wang, Y Chen, F Ye, in *Proc. of ACM MobiCom*. Push the limit of WiFi based localization for smartphones (ACM, New York, USA, 2012), pp. 305–316
21. P Mohan, V Padmanabhan, R Ramjee, in *Proc. of ACM SenSys*. Nericell: rich monitoring of road and traffic conditions using mobile smartphones (ACM, New York, USA, 2008), pp. 323–336
22. J Eriksson, L Girod, B Hull, R Newton, S Madden, H Balakrishnan, in *Proc. of ACM MobiSys*. The pothole patrol: using a mobile sensor network for road surface monitoring (ACM, New York, USA, 2008)
23. C Bo, L Zhang, X-Y Li, Q Huang, Y Wang, in *Proc. of ACM MobiCom*. SilentSense: silent user identification via touch and movement behavioral biometrics (ACM, New York, USA, 2013)
24. C Bo, L Zhang, T Jung, J Han, X-Y Li, Y Wang, in *Proc. of IEEE IPCCC*. Continuous user identification via touch and movement behavioral biometrics (IEEE, Piscataway, New Jersey, USA, 2014)
25. X Su, H Tong, P Ji, Activity recognition with smartphone sensors. Tsinghua Sci. Technol. **19**(3), 235–249 (2014)
26. JR Kwapisz, GM Weiss, SA Moore, Activity recognition using cell phone accelerometers. ACM SIGKDD. **12**(2), 74–82 (2011)
27. C Bo, X Jian, X-Y Li, X Mao, Y Wang, F Li, in *Proc. of ACM MobiCom*. You're driving and texting: detecting drivers using personal smart phones by leveraging inertial sensors (ACM, New York, USA, 2013)
28. C Bo, X Jian, T Jung, J Han, X-Y Li, X Mao, Y Wang, Detecting driver's smartphone usage via non-intrusively sensing driving dynamics. IEEE Internet of Things Journal. to appear
29. J Paek, J Kim, R Govindan, in *Proc. of ACM MobiSys*. Energy-efficient rate-adaptive GPS-based positioning for smartphones (ACM, New York, USA, 2010), pp. 299–314
30. C Bo, X-Y Li, T Jung, X Mao, Y Tao, L Yao, in *Proc. of ACM MobiCom*. SmartLoc: Push the limit of the inertial sensor based metropolitan localization using smartphone (ACM, New York, USA, 2013)
31. R Levi, T Judd, Dead reckoning navigational system using accelerometer to measure foot impacts, Dec 10 1996, US Patent 5,583,776
32. P Enge, A Van Dierendonck, Wide area augmentation system. Prog. Astronaut. Aeronaut. **164**, 117–142 (1996)
33. B Parkinson, P Enge, Differential GPS Global Positioning. Syst. Theory Appl. **2**, 3–50 (1996)
34. P Bahl, V Padmanabhan, in *Proc. of IEEE INFOCOM*. RADAR: An in-building RF-based user location and tracking system, vol. 2 (IEEE, Piscataway, New Jersey, USA, 2000), pp. 775–784
35. M Youssef, A Agrawala, The Horus location determination system. Wireless Netw. **14**(3), 357–374 (2008)
36. L Ni, Y Liu, Y Lau, A Patil, Landmarc: indoor location sensing using active RFID. Wireless Netw. **10**(6), 701–710 (2004)
37. Y Chen, D Lymberopoulos, J Liu, B Priyantha, in *Proc .of ACM MobiSys*. FM-based indoor localization (ACM, New York, USA, 2012), pp. 169–182
38. M Azizyan, I Constandache, R Roy Choudhury, in *Proc. of ACM MobiCom*. Surroundsense: mobile phone localization via ambience fingerprinting (ACM, New York, USA, 2009), pp. 261–272
39. S Sen, B Radunovic, R Choudhury, T Minka, in *Proc. of ACM MobiSys*. You are facing the Mona Lisa: spot localization using phy layer information (ACM, New York, USA, 2012), pp. 183–196
40. W Griswold, P Shanahan, S Brown, R Boyer, M Ratto, R Shapiro, T Truong, Activecampus: experiments in community-oriented ubiquitous computing. Computer. **37**(10), 73–81 (2004)
41. G Ananthanarayanan, M Haridasan, I Mohomed, D Terry, CA Thekkath, in *Proc. of ACM MobiSys*. Startrack: a framework for enabling track-based applications (ACM, New York, USA, 2009)
42. S Lanzisera, D Lin, K Pister, in *Proc. of IEEE WISES*. RF time of flight ranging for wireless sensor network localization (IEEE, Piscataway, New Jersey, USA, 2006), pp. 1–12
43. N Priyantha, A Chakraborty, H Balakrishnan, in *Proc. of MobiCom*. The cricket location-support system (ACM, New York, USA, 2000), pp. 32–43
44. D Niculescu, B Nath, in *Proc. of IEEE INFOCOM*. Ad hoc positioning system (APS) using AoA, vol. 3 (IEEE, Piscataway, New Jersey, USA, 2003), pp. 1734–1743
45. Z Zhang, X Zhou, W Zhang, Y Zhang, G Wang, B Zhao, H Zheng, in *Proc. of ACM MobiCom*. I am the antenna: accurate outdoor AP location using smartphones (ACM, New York, USA, 2011), pp. 109–120
46. A Harter, A Hopper, P Steggles, A Ward, P Webster, The anatomy of a context-aware application. Wireless Netw. **8**(2), 187–197 (2002)
47. L Girod, M Lukac, V Trifa, D Estrin, in *Proc. of ACM SenSys*. The design and implementation of a self-calibrating distributed acoustic sensing platform (ACM, New York, USA, 2006), pp. 71–84
48. G Borriello, A Liu, T Offer, C Palistrant, R Sharp, in *Proc. of ACM MobiSys*. Walrus: wireless acoustic location with room-level resolution using ultrasound (ACM, New York, USA, 2005), pp. 191–203
49. R Nandakumar, K Chintalapudi, V Padmanabhan, in *Proc. of ACM MobiCom*. Centaur: locating devices in an office environment (ACM, New York, USA, 2012), pp. 281–292
50. A Rai, K Chintalapudi, V Padmanabhan, R Sen, in *Proc. of ACM MobiCom*. Zee: Zero-effort crowdsourcing for indoor localization (ACM, New York, USA, 2012), pp. 293–304
51. D Titterton, J Weston, D Titterton, J Weston, *Strapdown inertial navigation technology*. (The Institution of Engineering and Technology, Michael Faraday House, Six Hills Way, Stevenage, Herts. SGl 2AY, UK, 2004)

52. R Jirawimut, P Ptasinski, V Garaj, F Cecelja, W Balachandran, A method for dead reckoning parameter correction in pedestrian navigation system. Instrum. Meas. **52**(1), 209–215 (2003)
53. O Woodman, R Harle, in *Proc. of ACM Ubicomp*. Pedestrian localization for indoor environments (ACM, New York, USA, 2008), pp. 114–123
54. M Youssef, M Yosef, M El-Derini, in *Proc. of IEEE GLOBECOM*. GAC: energy-efficient hybrid GPS-accelerometer-compass GSM localization (IEEE, Piscataway, New Jersey, USA, 2010), pp. 1–5