

# A Data-driven Inertial Navigation/Bluetooth Fusion Algorithm for Indoor Localization

Jianfan Chen, Baoding Zhou, Shaoqian Bao, Xu Liu, Zhining Gu,  
Linchao Li, Yangping Zhao, Jiasong Zhu, and Qingquan Li

**Abstract**—The introduction of data-driven inertial navigation provides new opportunities that the pedestrian dead reckoning could not well provide for constraining inertial system error drift on smartphones, and has been considered as another promising approach to meet the requirement of location-based services. However, indoor localization systems based on a single technology still have their limitations, such as the drift of inertial navigation and the received signal strength fluctuation of Bluetooth, making them unable to provide reliable positioning. To exploit the complementary strengths of each technology, this paper proposes a feasible fusion framework by utilizing a particle filter to integrate data-driven inertial navigation with localization based on Bluetooth Low Energy (BLE). For data-driven inertial navigation, under the premise of using the deep neural network with great potential in model-free generalization to regress pedestrian motion characteristics, we effectively combined the method of using gravity to stabilize inertial measurement units data to make the network more robust. Experimental results show that in the test of different smartphone usages, the proposed data-driven inertial navigation and BLE-based localization technology have good results in modeling user's movement and positioning respectively. And due to this, the proposed fusion algorithm has almost unaffected by the usages of smartphones. Compared with BLE-based localization that achieved a good mean positional error (MPE) of 1.76m, for the four usages of texting, swinging, calling and pocket, the proposed fusion algorithm reduced the MPE by 32.35%, 20.51%, 20.74%, and 45.37%, respectively, and can further improve localization accuracy on the basis of existing fusion method.

**Index Terms**—indoor localization, smartphone sensor, data-driven inertial navigation, Bluetooth Low Energy, particle filter

This work was supported in part by the National Key R&D Program of China (2016YFB0502204), Guangdong Basic and Applied Basic Research Foundation(2019A1515011910), Shenzhen Scientific Research and Development Funding Program (JCYJ20190808113603556, KQTD20180412181337494), National Natural Science Foundation of China (61901281). (*Corresponding author: Baoding Zhou*)

Jianfan Chen is with the College of Electronics and Information Engineering, Guangdong Key Laboratory of Urban Informatics, Shenzhen Key Laboratory of Spatial Smart Sensing and Services, and MNR Key Laboratory for GeoEnvironmental Monitoring of Great Bay Area, Shenzhen University, Shenzhen 518060, China (e-mail: 1800261033@email.szu.edu.cn).

Baoding Zhou, Linchao Li, Yangping Zhao, and Jiasong Zhu are with the College of Civil and Transportation Engineering, Shenzhen University, Shenzhen 518060, China, and Institute of Urban Smart Transportation & Safety Maintenance, Shenzhen University, Shenzhen 518060, China (e-mail: bdzhou@szu.edu.cn, lilinchao@szu.edu.cn, yangping@szu.edu.cn, zjsong@szu.edu.cn).

Shaoqian Bao is with Shenzhen Escope Intelligence technology co. LTD, Shenzhen, China. (e-mail: baoshaoqian@hotmail.com).

Xu Liu and Qingquan Li are with the College of Civil and Transportation Engineering, Shenzhen University, Shenzhen 518060, China (e-mail: xuliuk\_ksy@126.com, liqq@szu.edu.cn).

Zhining Gu is with the School of Architecture and Urban Planning, Shenzhen University, Shenzhen 518060, China (e-mail: 1900325009@email.szu.edu.cn).

## I. INTRODUCTION

INDOOR localization has attracted much interests in recent years due to the diverse location-based services (LBS) that require accurate positioning [1], [2]. The outdoor positioning system based on Global Position System (GPS) has achieved good results, however, since the accuracy of satellite system is degraded in indoor environments, researchers have tried multiple technologies suitable for indoor localization that potentially can be implemented on smartphones, e.g., WiFi [3], [4], radio-frequency identification (RFID) [5], Bluetooth [6], indoor map [7], [8] and inertial sensors-based localization [9]. There is still no single technology that can provide reliable indoor positioning [10]. The character of each technology determines its applicable indoor positioning occasions, highlighting their advantages and disadvantages.

Unbounded system drift has always been a problem that plagues inertial sensors-based localization [11]. Performing inertial navigation and localization on smartphones has long been a research focus, whose task is to use measurements provided by inertial measurement units (IMU) for navigation and localization. Indeed, with the recent advances of micro-electro-mechanical systems (MEMS), most modern smartphones are equipped with small, cheap and efficient IMUs. But it is precisely due to these properties, inertial navigation based on smartphones is plagued by sensor noise and bias. And the propagation of orientation errors caused by sensor noise perturbing the gyroscope signals eventually becomes the main reason for the drift of traditional strapdown inertial navigation system (SINS) [9]. To address this problem, a solution by combining inertial sensors with a video camera as visual-inertial-odometry (VIO) has been proposed [12], [13]. While being successful, the addition of a video camera adds extra cost and battery drain. Another high-profile solution is step-based pedestrian dead reckoning (PDR). Different from strapdown inertial navigation system, step-based PDR uses inertial measurement to perform the following tasks [14]: step count or step segmentation, step length estimation and step direction estimation. Through the step length update, although the step-based PDR can effectively compensate the drift of the inertial system, the user's walking habits seriously affect the dynamic step length estimation. Furthermore, the assumption of periodicity used in step detection are limited and may fail in some specific situations (*e.g. phone on a trolley*).

Data-driven approach, because of its superior mobility and flexibility, has been considered as another promising approach to constrain inertial system error drift. In order to achieving

more competitive performance, it utilizes IMU sensors data in a short time and ground-truth motion trajectories to regress motion parameters (*e.g. velocity and heading*). Several devices, such as Vicon [15], Apple ARKit [16] and Google Tango [17], have been used to obtain the ground-truth motion trajectories for training. Robust IMU double integration (RID) has made a breakthrough in the coordinate frame normalization and successfully regressed a relatively accurate velocity vector [17]. Moreover, influenced by the great potential for model-free generalization of deep neural network (DNN), IONet focuses on using Long Short-Term Memory (LSTM) to regress the velocity and heading rate of pedestrian motion [15]. In addition to using data-driven approach for pedestrian motion estimates, what's more interesting is that, for vehicle motion, AI-IMU even trained a DNN to estimate IMU noise parameters and perform localization by using an Extended Kalman Filter [18]. Backprop Kalman Filter also designed neural network architectures to optimize the task of vehicle state estimation [19]. These data-driven methods convert inertial tracking into a sequential learning problem, which is superior to traditional inertial navigation systems in terms of robustness.

However, like common inertial navigation methods, data-driven inertial navigation methods also have serious disadvantages: (a)when the starting point is unknown, inertial navigation can only obtain relative coordinates but not absolute coordinates in the global coordinate frame; (b)as the inertial tracking process continues, the system will inevitably be plagued by drift. Therefore, a series of technologies such as WiFi and Bluetooth have been adopted accordingly to meet the challenges of inertial navigation. In particular, due to the simplicity, reliability and efficiency of Bluetooth, many indoor localization methods use Bluetooth Low Energy (BLE)'s Received Signal Strength Index (RSSI) for accurate positioning. This approach does not require any specialized hardware or additional infrastructure support because most smartphones are Bluetooth enabled. Moreover, extensive experiments performed by Zhao et al. shows that under the same environment and conditions, BLE-based localization is more accurate than WiFi by around 27 percent [20]. BLE-based indoor localization methods are mainly divided into two categories: ranging-based and non-ranging-based. The ranging-based methods mainly include triangulation [21], and the non-ranging-based methods mainly include fingerprint algorithms [22], [23]. Although both methods address the challenges of inertial navigation, the positioning error is still large due to the multipath effect and shadowing caused by obstacles in the indoor environments, so positioning by Bluetooth alone cannot well meet the needs of LBS.

A particular interest is to fuse the above two technologies: inertial sensors-based localization technology and BLE-based localization technology, as these two technologies are complementary to each other. A fusion localization method based on multiple positioning technologies can not only exert the advantages of each technology, but also inhibit the disadvantages. Filter theories such as Kalman Filters (KF) [24], Extended Kalman Filters (EKF) [25], [26], and Unscented Kalman Filters (UKF) [27] have been widely used to fuse different data. EKF and UKF are optimizations of KF. Taking into

account the limitation of accuracy that the first-order Markov hypothesis of EKF is relatively simple [28], while UKF only uses a few sigma points to approximate the probability density distribution, a particle filter based on Bayesian statistical theory and sequential Monte Carlo framework is selected [29]. The particle filter approach can approximate any probability density function [28], and this method with various modifications has been used in [30]–[32], among others. When using particle filter to fuse the above two technologies, the estimated performance of positioning is directly affected by the results of BLE-based localization, and the estimation accuracy of motion characteristics (distance and heading). In order to promote the fusion performance, recent innovations have proposed ways to improve BLE-based localization or distance estimation with high precision, which used regression model to facilitate BLE-based localization or utilized Channel State Information (CSI) to promote moving distance estimation [30], [31]. However, they still follow standard operations (obtain heading directly through gyroscope integration or magnetometer readings) to derive orientation information, which will cause large error of heading estimation [33]. Although the floorplan added by EasiTrack can effectively limit such heading errors, it requires additional map information and increases the complexity of the fusion system. Fortunately, the introduction of data-driven inertial navigation provides new possibilities for constraining inertial system error drift (not limited to constrain the error of heading estimation), giving us the opportunity to utilize it in the fusion algorithm to estimate pedestrian motion characteristics, and verify its help for accuracy improvement through experiments. This paper focuses on researching the integration of data-driven inertial navigation and BLE-based localization with better practicability and seeks to take indoor localization research to the next level via the following three contributions:

- We proposed a feasible method that utilizes particle filtering to fuse data-driven inertial navigation and BLE-based localization for indoor positioning. Due to the integration of data-driven inertial navigation, the proposed fusion algorithm is almost unaffected by different usages of smartphones, and can further improve localization accuracy on the basis of existing fusion method.
- We effectively combined the approach (called coordinate frame alignment) that uses gravity to stabilize IMU data, making the network more robust to the regression of pedestrian motion characteristics. Especially for heading regression, the smaller regression error in the proposed method well verifies the superiority of introducing coordinate frame alignment module.
- A dataset with visual-inertial odometry based ground truth and IMU sensor measurements across multiple device placements. We will share it publicly to facilitate further research.

The rest of the paper is organized as follows. After this introduction, we review some related work in data-driven inertial navigation systems, BLE-based localization and fusion localization methods in Section II. In Section III, the method and localization algorithms presented in details. Section IV provides experimental results. Finally, Section V concludes

this paper.

## II. RELATED WORK

In this section, we briefly outline some related work in data-driven inertial navigation systems, BLE-based localization and fusion localization methods.

**Data-driven inertial navigation:** The data-driven technology has been increasingly used in many recent works, but these systems focus on gait pattern recognition rather than estimation of walking speed [34]–[36]. DeepWalking is a representative system that uses data-driven technology to achieve speed regression at the beginning of this research phase [37]. Although it can only utilize the sensors data on a treadmill to regress speed scalar, it is also beneficial to the study of data-driven inertial navigation. Several solutions for pedestrian walking speed estimation are actually subject to various limitations such as not being applicable to mobile-based applications, and lack of consideration for users to use smartphones in different usages [38]–[40]. RIDI is the first data-driven inertial navigation system to estimate speed in a real indoor environment [17]. In principle, its breakthrough in coordinate frame normalization and the use of support vectors make it regress a more accurate velocity vector. In addition, influenced by the great potential for model-free generalization of DNN, IONet utilizes LSTM to regress the magnitude of speed and the rate of change of heading [15]. Recent innovations have promoted data-driven inertial navigation to accurately achieve three-dimensional inertial navigation, which underpins the foundation of realizing cross-floor indoor positioning [16], [41]. Other prospective studies even used data-driven methods to estimate vehicle pose in the KITTI odometry dataset [18], [19], [42]. Between them, AI-IMU only used IMU data to perform dead-reckoning of wheeled vehicles, deducing data-driven inertial navigation in a scene different from pedestrian motion estimation [18].

**BLE-based localization:** In the past few years, BLE-based localization systems have been widely used. Currently popular BLE on the market are Apple's iBeacon and Google's URIBeacon. Bluetooth was not designed for positioning, but it can be used to estimate user location by utilizing RSS measurements. Though the indoor localization methods based on BLE and WiFi are basically the same, the research by Zhao et al. shows that under the same environment and conditions, BLE-based localization is still more accurate than WiFi because Bluetooth has lower transmission power and a unique channel hopping mechanism [20]. Another advantage of BLE is that it can be placed freely and provides good signal geometry [43]. Many studies have focused on the most typical BLE-based localization method, the trilateration, which utilizes the geometry of the circle to calculate to user's location [21], [44]. Moreover, the fingerprint algorithm is another alternative method for BLE-based localization [22]. Pavel et al. used conventional flat weighted k-nearest neighbor (WKNN) algorithm to perform BLE-based localization well [45]. WKNN sorts the database fingerprints according to the Euclidean distance to online RSSI, and then selects the top  $k$  fingerprints for weighted estimation of location. Compared

with the fingerprint algorithm, the trilateration method is limited by the application scenario, so its positioning accuracy is usually unstable [46].

**Fusion localization:** Since no single technology can provide reliable indoor localization, some researchers have attempted to find solutions by integrating multiple positioning technologies [10]. Studies have shown that the KF and particle filter can be used to promote GPS positioning, so they can also be used to improve the performance of indoor localization systems [47]. For instance, references showed the high performance of KF, EKF and UKF in indoor localization [24], [25], [27], [48]. However, the limitations of EKF and UKF for solving nonlinear problems make particle filter, which can approximate any probability density function, stand out in the indoor positioning and be considered in this paper [28]. If the method proposed by EA Wan et al. [49], which used Kalman filter in velocity estimation and particle filter in orientation estimation, cannot be regarded as a representative of completely using particle filter for fusion, then the proposal of EasiTrack completely proves the excellent performance of the particle filter in indoor positioning [30]. On the other hand, the estimated performance of fusion method is directly affected by the results of RF-based localization, and the estimation accuracy of distance and heading. Some existing fusion methods use regression model to facilitate RF-based localization or utilize CSI to promote distance estimation, but still obtain heading directly through gyroscope integration or magnetometer readings, resulting large error of heading estimation. These methods default to the errors caused by traditional orientation estimation during fusion, however, in order to improve the accuracy under different postures, methods of attitude detection or floorplan are cumbersomely introduced [25], [30]. The others even make the system very complex and seem to be unsuitable for finite resources such as telephones [50]. The finite computing power, battery capacity and storage capacity of telephones are the main manifestations of their finite resources. With the continuous improvement of the computing performance of smartphones, complex fusion positioning algorithms such as Monte Carlo localization of mobile robots, simultaneous localization and mapping (SLAM) may gradually appear in the field of smartphone-based indoor localization [51], [52]. But for now, fusion positioning can only take up a small part of the CPU processing time to ensure the normal operation of other mainstream functions. From the perspective of energy saving, it is impossible to keep the smartphone in a continuous high-speed operation state, otherwise the battery of the mobile phone will be exhausted quickly. In addition, the limited storage capacity on smartphones makes it insufficient to run complex deep learning fusion algorithms. Using the data collected by phones, this paper implemented the proposed method on a computer. After the fusion method is transplanted into phones, the impact analysis on smartphone resources will become our focus.

## III. METHODOLOGY

The architecture of the proposed fusion approach for indoor localization on a smartphone is illustrated in Fig. 1. Our approach consists of three key components: Data-driven Inertial

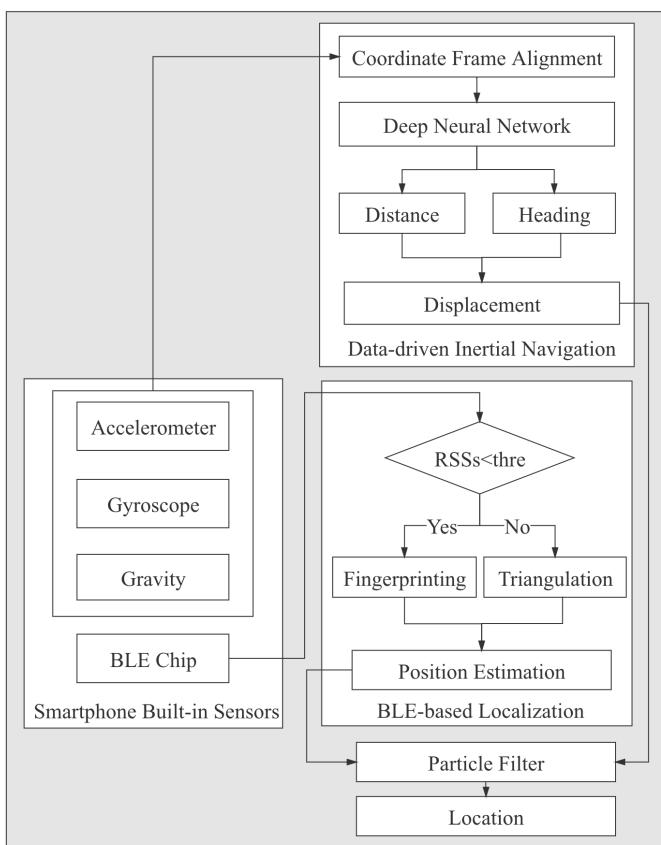


Figure 1: Overview of the fusion approach

**Navigation, BLE-based Localization and Particle Filter.** The data-driven inertial navigation module utilizes inertial sensors and trained DNN to predict distance and heading of pedestrian motion. The BLE-based localization module compares the received signal strength (RSS) with the threshold to choose a more suitable method for BLE-based localization. And then, the particle filter module receives the motion vector and fuses it with the BLE-based localization result, and finally outputs localization result. More details on each module are described in the following sections.

#### A. Data-driven Inertial Navigation

The consecutive IMU data in a short time contains enough information to represent body motions [53]. Very recently, some papers have successfully used such IMU data to regress velocity [15]–[17], [37]. Combining RIDI’s breakthrough in the coordinate axis method and the convenience of IONet’s use of RNN to model motion, the data-driven inertial navigation method proposed in this paper contains three key steps: 1) Acquire the training data necessary for the sequential learning, 2) Extract phone orientation independent data by coordinate frame alignment and 3) Utilize deep neural network to learn the relationship between data features and motion characteristics for distance and heading regression. We now explain the details of the three steps.

1) *Data acquisition:* Our inertial navigation approach is data-driven with supervised learning. More specifically, it

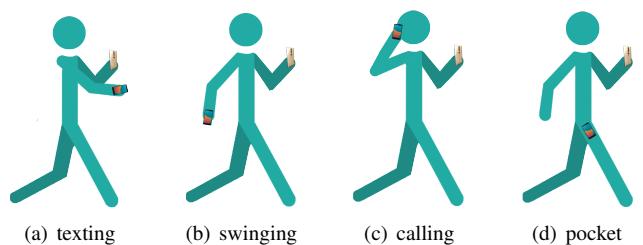


Figure 2: Data acquisition under four basic usages

utilizes ground-truth motion trajectories and IMU sensors data to regress motion parameters. Assuming the subject is walking on a flat ground, the regression of the motion parameters is performed on a 2D horizontal plane. Based on observations of daily smartphone usage habits, four basic usages of smartphones in this paper are considered (*i.e.*, *texting*, *calling*, *swinging* and *in the pocket*). We allow subjects to hold a Tango with one hand to capture the ground-truth of the motion trajectory, and meanwhile, carry a Samsung Galaxy S8 in one of the ways mentioned above to capture authentic IMU sensors data from the smartphone (see Fig. 2).

For ground-truth motion trajectory, we refer to the method of RIDI to capture 3D camera poses from the Visual Inertial Odometry system on Tango (<https://github.com/higerra/TangoIMURecorder>) [17]. As a representative application of Visual-inertial SLAM (Vi-SLAM), Tango combines V-SLAM and IMU sensors to resolve the ambiguity of the scale and produce motion data [54], [55]. Under the condition of clear vision, Tango can provide a low-drift motion trajectory (after tracking 200 meters, the position error is less than 1 meter), which can offer pseudo ground truth. For IMU sensors data, we implemented a data capture application for a S8 to record raw sensor data from the built-in phone sensors (*e.g.* *accelerometer*, *gyroscope* and *gravity*). Also note that these captured data are time-stamped by their respective applications. The TangoIMURecorder captures pose data at 200Hz. We synchronize the IMU sensors data into the time-stamps of Tango poses via linear interpolation.

2) *Coordinate Frame Alignment:* According to Newtonian mechanics, accelerometer and gyroscope data are inextricably linked to motion, so people naturally think of using them to learn to regress velocity vectors. However, people carry smartphones in various ways, making the built-in IMU sensor coordinate frame unstable. The arbitrary orientation of smartphone makes it hard to extrapolate reliable motion vectors using only accelerometer and gyroscope data. Stable IMU data features will make regression tasks easier. RIDI therefore proposes a stabilized coordinate frame, extracting phone orientation independent data from IMU sensors data.

More specifically, the stabilized coordinate frame introduces gravity measurements to eliminate the pitch and roll ambiguities of a smartphone. We record the raw accelerometer, gyroscope and gravity measurements via the Android API. By aligning the y-axis of the phone with the direction of negative gravity, the accelerometer and gyroscope data can be express

on a fixed coordinate frame (see Fig. 3). To suppress high-frequency noise, we apply Gaussian smoothing with  $\sigma = 2.0$  samples to the aligned accelerometer and gyroscope data [17]. In addition, because Tango provides the pose data in the global coordinate frame from the Android API, it is necessary to convert the velocity vector to the stabilized coordinate frame. Assume that the user and the smartphone are aligned to the same coordinate frame. Then when the user is walking on a flat ground, that is, the vertical displacement is zero, we can convert the pose data into stabilized heading and displacement. These aligned accelerometer and gyroscope data are integrated with motion characteristics to construct the training data for DNN.

3) *Distance and Heading Regression*: On the basis of IONet [15], the prediction is performed by a recurrent neural network (RNN) and this section explains the proposed framework of the RNN (see Fig. 4). The input to the network is the IMU data features (*i.e.*, aligned accelerometer and gyroscope data) over a few seconds and the output is the motion characteristics (*i.e.*, distance and heading). As mentioned above, we use the accelerometer, gyroscope, gravity and 3D camera to extract these data in the coordinate frame alignment module. With the task of regressing the location distance  $\Delta l$  and heading change  $\Delta\psi$  over measurements of the finite window size of time  $n$ , the network learns the latent relationship between data features and motion characteristics, which can be expressed as:

$$(\Delta l, \Delta\psi) = \text{RNN}(\{(a_i, \omega_i)\}_{i=1}^n) \quad (1)$$

We adopt a two-layer Bi-directional LSTM framework as the core building block due to its ability to better utilize long-term dependencies [56]. Each LSTM has 256 hidden states. To alleviate the problem of overfitting, a dropout layer that randomly sets the input element to zero with a probability of 0.5 is placed after each LSTM layer. A fully connected layer is finally added to perform the regression of the location distance and heading change. The mean square error between the predicted motion vectors and the ground-truth is the value we need to minimize. We selected the

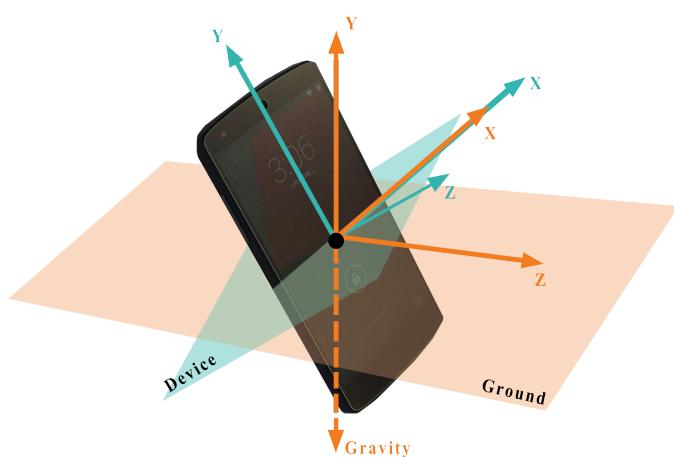


Figure 3: Coordinate frame alignment

ADAM optimizer to minimize this value and learn the optimal parameters inside the RNN [57]. And we use tensorflow (<https://tensorflow.google.cn/>) to implement the network.

When training the network, we must consider not only the accuracy, but also its practicality. Therefore, we experimented its validation results with input window sizes of 200, 400 and 600 frames (see Fig. 5). It can be seen from figure 5 that, compared with other window sizes, the input data of 200 frames seems insufficient for the network to well capture its latent relationship with motion characteristics. In addition, the input data of both 400 frames and 600 frames can reduce the validation loss of the network to a steady and low state, while the input data of 400 frames reduces the validation loss to a steady state at a faster rate. In order to achieving more competitive performance, we finally select the window size  $n$  as 400 frames (2 seconds), which means that the 2400-dimensional long-term dependent feature vector is constructed by stacking aligned accelerometer and gyroscope data. We feed this feature vector into the RNN to predict the distance changing  $\Delta l$  and heading changing  $\Delta\psi$  within 2 seconds. After obtaining the  $\Delta l$  and  $\Delta\psi$  in the stabilized coordinate frame, the next location can be expressed as:

$$\begin{cases} x = x_0 + \Delta l * \cos(\psi_0 + \Delta\psi) \\ y = y_0 + \Delta l * \sin(\psi_0 + \Delta\psi) \end{cases} \quad (2)$$

where  $(x_0, y_0)$  and  $\psi_0$  are the previous location and heading.

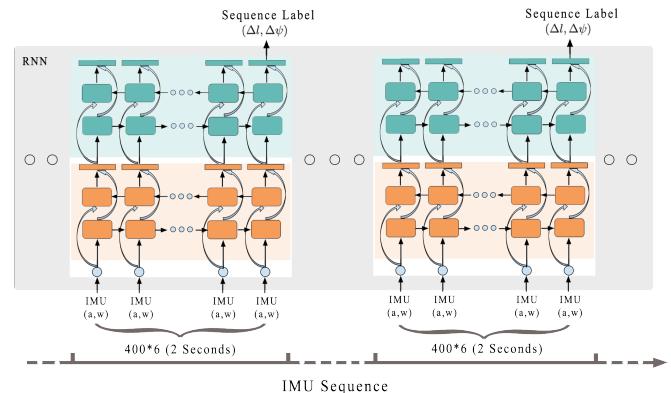


Figure 4: The RNN framework of the proposed method

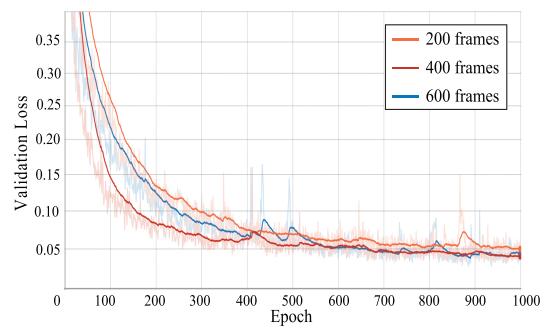


Figure 5: Loss of adopting various frames

### B. BLE-based Localization

Another technology fused by our approach is Bluetooth. Based on the characteristics of Bluetooth, a BLE-based localization approach suitable for implementation in indoor environment is presented in this section.

Although Bluetooth was not designed for positioning, it can be used to estimate users location by utilizing RSS measurements. A basic method to localization is trilateration. In most cases, the closer the device is to the BLE beacon, the larger the RSS measurement. Therefore, the RSS of the BLE beacon can reflect the distance between the device and the beacon, which can be expressed as [58]:

$$RSS(\lambda) = RSS(\lambda_0) - 10\eta \log \left( \frac{\lambda}{\lambda_0} \right) + X_\sigma \quad (3)$$

where  $\lambda$  is the distance from the beacon to device,  $RSS(\lambda)$  is RSS of a beacon,  $RSS(\lambda_0)$  is RSS (dBm) at reference distance  $\lambda_0$ ,  $\eta$  is the path loss exponent, and  $X_\sigma$  is the zero-mean Gaussian noise with variance  $\sigma^2$ . By converting several (three or more) RSS measurements to the distance from the device to the beacon, the trilateration method uses the geometry of the circle to calculate the position of the device (see Fig.6). More specifically, on the premise of knowing the location of the BLE beacons, assuming that the location of the device is  $(x, y)$  and the location of the beacon is  $(x_i, y_i)$ , we can get  $N$  equations:

$$(x - x_i)^2 + (y - y_i)^2 = \lambda_i^2, i = 1, 2, \dots, N \quad (4)$$

By solving the  $N$  equations, we can obtain the location of the device. Under ideal condition (see Fig. 6a), the three circles intersect at a point, and we can get the unique solution of the equation. However, in complex indoor environments, the Bluetooth signals are unstable because they can be reflected by walls and attenuated by human bodies. Due to the fluctuation of the RSS of the BLE, the three circles usually do not intersect at one point, but may or may not intersect in a region (see Fig. 6b and 6c). In such imperfect situations, the Line Intersection-based Trilateration method can be utilized to find the approximate solution of the equation, so as to obtain the BLE-based localization result [59].

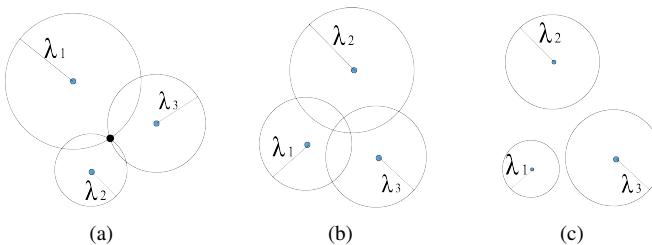


Figure 6: Trilateration method

In fact, though the trilateration method can estimate position under imperfect situations, it does not solve the situation in Fig. 6c well because the distance reflected by RSS with a small value is usually not reliable. Naturally, we think of another alternative way to explicit conversion of signal strength measurement into distance, which is to compare

RSS measurements with a radio map. This method is called location fingerprinting. BLE fingerprinting-based positioning method consists of two phases: offline training phase and online positioning phase. In the training phase, a set of known locations are selected as the reference points (RPs) and RSSs from all detected BLE beacons are collected at each RP. The RSSs collected at each RP is called fingerprints. The fingerprints of all the RPs form a radio map. In the online phase, the real-time RSS samples received from the beacons are compared to the stored radio map to estimate the position.

The BLE-based localization module compares the RSS with a threshold to choose a more suitable method for BLE-based localization. We set the threshold to -50 dBm based on experience. If the RSS values of all observed BLE beacons are less than the threshold, we use a fingerprinting method to calculate the location; otherwise, we utilize triangulation method to estimate the location. In this way, when the RSS values of all observed BLE beacons are very small, the introduction of fingerprinting method can greatly improve the estimated positioning results using these RSS values for triangulation. It is worth noting that because the distance-power gradient is relatively small, the use of RSS to estimate the wireless distance and fingerprint has great uncertainty. In other words, the Bluetooth positioning estimation methods used in this article may not be as accurate as the the radio signal-based estimation that utilizes the angle of arrival or time delay measurement [10]. Instead of focusing on BLE-based positioning, this article attempts to implement a feasible way to integrate data-driven inertial navigation with BLE-based positioning.

### C. Particle Filter

Particle filter is based on a sequential Monte Carlo framework, which utilizes a set of weighted random particles to represent the posterior density of unknown positions in the dynamic state estimation framework [29]. We use particle filter to fuse the above two technologies: data-driven inertial navigation technology and BLE-based localization technology. The device positions obtained by BLE-based localization are used as observation data, and the data-driven inertial navigation is used to model the user's movement.

**1) Initialization:** Suppose the set of particles is  $H = \{X^i | i = 1, 2, \dots, N\}$ ,  $N$  is the number of particles. Each particle has a three-dimensional joint probability distribution, (*i.e.*,  $X^i = (x^i, y^i, \theta^i)$ ), where  $(x^i, y^i)$  represents the 2D position of the  $i$ th particle and  $\theta^i$  is its orientation. At time  $t = 0$ , using the Gaussian distribution, randomly select the position of the particles around the initial point  $(x_0^i, y_0^i)$  obtained by BLE-based localization. In addition, the initial weights  $\omega_0^i$  of each particle is equal to  $1/N$ .

**2) Particle Movement:** The introduction of data-driven inertial navigation provides new possibilities for constraining inertial system error drift, giving us the opportunity to utilize it in the fusion algorithm. Instead of using PDR for multiple particles, this article innovatively utilizes data-driven inertial

navigation to model the user's movement, which is follow as:

$$\begin{cases} x_t^i = x_{t-1}^i + l_t * \cos\psi_t \\ y_t^i = y_{t-1}^i + l_t * \sin\psi_t \end{cases} \quad (5)$$

where distance  $l$  and heading  $\psi$  at time  $t$  obtained by data-driven inertial navigation estimation may contain more precise information about the particle movement.

3) *Particle Weight Update and Resampling*: When a new observation data (*i.e.*, a new position) is obtained from BLE-based localization at time  $t$ , the weights of all particles need to be updated. The measurement update is applied based on how close the particle position is to the BLE-based localization result. And particle resampling is performed to retain the particles with greater weight and discard the particles with less weight.

4) *Target Location Estimation*: In order to make the posterior probability expression smoother and reflect the superiority of particle filtering, in this paper, the weighted sum of particles is used to estimate the final target location, which can be expressed as:

$$\begin{cases} X_t = \sum_{i=0}^N x_t^i \omega_t^i \\ Y_t = \sum_{i=0}^N y_t^i \omega_t^i \end{cases} \quad (6)$$

#### IV. EXPERIMENT AND RESULTS

To evaluate the proposed approach, we implemented the positioning methods in an office building, with a  $52.5m \times 52.5m$  floor plan (see Fig. 7). All beacons were mounted at a height of 2.5m above the ground with the same technical configuration, and the horizontal interval between beacons is approximately 3m. Considering that the concrete structure of the wall is not convenient for us to fix the beacon on it, we mounted the beacon on the ceiling, which also helped us to unify the mounted height. In addition, we used S8 to collect IMU sensors data in four specific usages, and used Tango device to provide pseudo ground truth. By collecting data from different subjects in the experimental scene, we collected 116 sequences. Considering the error of Tango device collecting ground truth, the length of each sequence is less than 200 meters. In the end, we have created one training sample per

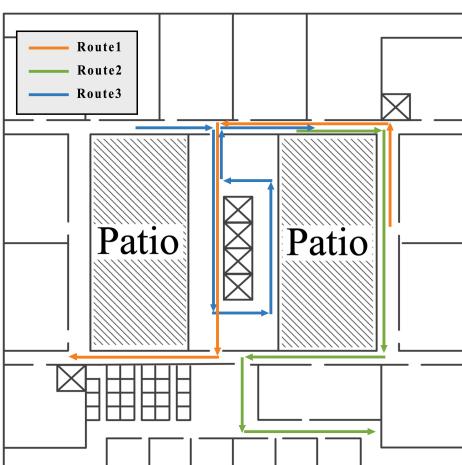


Figure 7: Experiment environment

Table I: Sequence parameters

|                             | Usage    | Seqs | Times(s) |
|-----------------------------|----------|------|----------|
| Training                    | Texting  | 25   | 5680     |
|                             | Swinging | 25   | 6248     |
|                             | Calling  | 25   | 5664     |
|                             | Pocket   | 25   | 5464     |
| Testing Data (Random Route) | Texting  | 1    | 26       |
|                             | Swinging | 1    | 34       |
|                             | Calling  | 1    | 22       |
|                             | Pocket   | 1    | 20       |
| Testing Data (Fixed Route)  | Texting  | 3    | 230      |
|                             | Swinging | 3    | 242      |
|                             | Calling  | 3    | 242      |
|                             | Pocket   | 3    | 242      |
| Total                       |          | 116  | 24114    |

100 IMU samples, resulting in a total of 23056 seconds of data, which can be used for network training. And a total of 1058 seconds of data, which is divided into sequences of random routes and fixed routes, to test its performance. Specifically, we chose three representative fixed routes: 1) Route 1 passes through the corridor and open area, 2) Route 2 includes corridors and indoor office areas (with many pedestrians), 3) Route 3 goes around the elevators, so the obstruction of the elevator will affect the data collection on this route (see Fig. 7). The detailed sequences parameters are listed in Table I.

#### A. Performance of Data-driven Inertial Navigation

The contribution of the fusion localization algorithm we proposed is that we introduced data-driven inertial navigation technology to model the user's movement, which provides new opportunities that the PDR could not well provide for constraining inertial system error drift on smartphones. As mentioned earlier, we combine the advantages of the RIDI and IONet methods. To be precise, unlike the IONet method that uses raw IMU data to train the network, our data-driven method uses the coordinate frame aligned IMU data for network training to make our method efficient without losing its accuracy. To evaluate the performance of the proposed data-driven method, we implemented two experiments: 1) By asking the subjects to walk randomly in the experimental environment, we evaluated the inertial navigation performance of the proposed data-driven method at short distances and 2) By asking the subjects to walk along three fixed routes (see Fig. 7), we evaluated the accuracy of the proposed data-driven method in regressing distance and heading.

1) *Inertial Navigation Performance of Data-driven Method at Short Distances*: As the inertial tracking process continues, the inertial navigation method will inevitably be plagued by drift. Therefore, we first evaluated the inertial navigation performance of the proposed data-driven method at short distances. We asked the subjects to walk randomly in the experimental environment to collect IMU sensor data with S8 in four usages, and then we used the trained DNN to predict distance and heading of subjects motion. We compared our data-driven method against a traditional PDR method [60] and IONet, and Table II summarizes the quantitative evaluations on the accuracy over 4 testing sequences. The third and fourth columns are the mean absolute errors (MAE) of the regressed

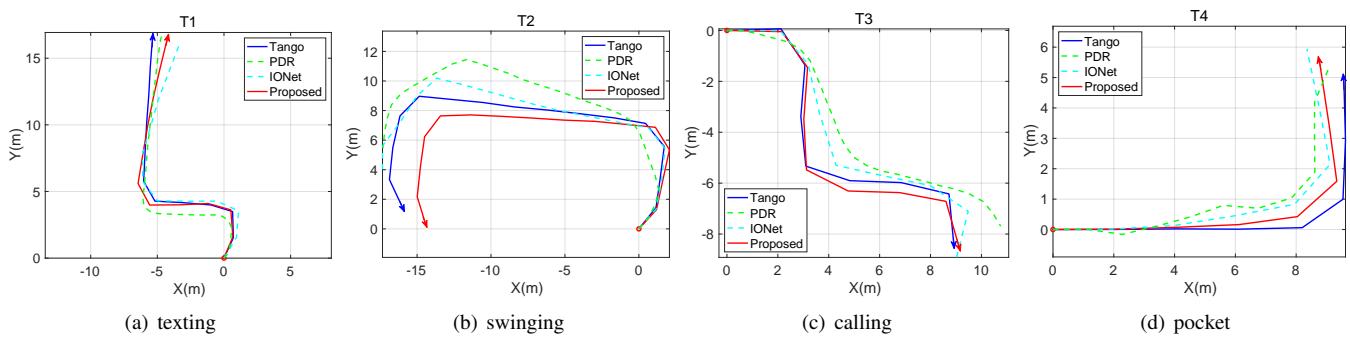


Figure 8: Trajectory reconstruction results

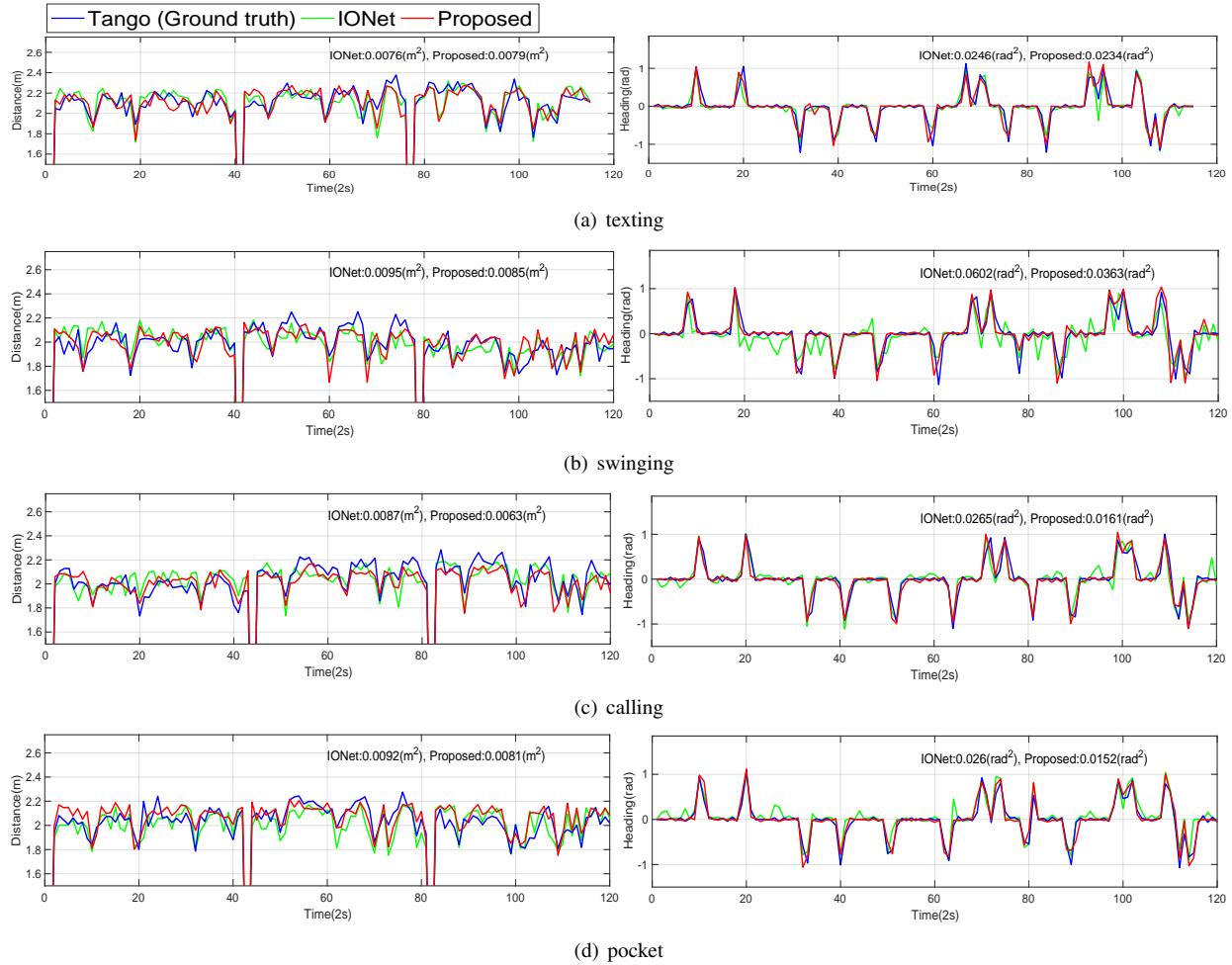


Figure 9: The distance and heading estimation for four usages under three fixed routes

distance in meter and heading in rad. Last three columns are the mean positional errors (MPE) in meter.

From Table II we can see that the mean absolute errors on the regressed motion characteristics of both data-driven methods are small. And in all sequences, the heading regression performance of the proposed method is superior to the method of IONet. This is because the coordinate frame alignment module improves the stability of heading regression, which is very important for trajectory reconstruction. Table II also shows that the mean positional errors of both data-

driven methods for all the routes were less than that of PDR. Furthermore, in order to more intuitively explain the superiority of the data-driven method, using the distance and heading obtained by regression and setting the origin as the starting point, we illustrate the generated trajectories, as shown in Fig. 8. Trajectories constructed by PDR, IONet methods and the ground truth are also displayed in Fig. 8. It can be seen that the trajectories reconstructed by our method works well in every usage. The error caused by low-cost sensors has always been a problem that plagues inertial navigation.

Table II: The motion characteristics and position evaluation

| Seq. | Usage    | Reg.Err.<br>(IONet) | Reg.Err.<br>(Proposed) | MPE<br>(IONet) | MPE<br>(Proposed) | MPE<br>(PDR) |
|------|----------|---------------------|------------------------|----------------|-------------------|--------------|
| T1   | Texting  | (0.0990,0.0917)     | (0.0886,0.0439)        | 0.5721         | 0.3809            | 0.7026       |
| T2   | Swinging | (0.1091,0.0680)     | (0.1198,0.0606)        | 1.0004         | 1.2550            | 1.6358       |
| T3   | Calling  | (0.0669,0.1332)     | (0.0396,0.0827)        | 0.6750         | 0.2143            | 1.0816       |
| T4   | Pocket   | (0.0658,0.0409)     | (0.0838,0.0333)        | 0.7066         | 0.3792            | 0.8663       |

Several solutions have been proposed that are more dedicated to improving distance estimation or combining with maps to prevent the accumulative errors of PDR [61], [62]. However, data-driven methods convert inertial tracking into a sequential learning problem, providing new possibilities for constraining inertial system error drift. In other words, this provides an additional possibility to improve the accuracy of the heading estimation for the method that only improves the accuracy of the distance estimation. By analyzing these trajectory results from Fig. 8, we can intuitively find that data-driven inertial navigation can achieve results close to or better than PDR in the test of four usages. That is, the data-driven method is superior to the traditional inertial navigation method in terms of robustness. But as the distance increases, the inertial tracking still drifts, so we have integrated Bluetooth technology and will show the results of fusing different inertial navigation technologies in later experiments. More interestingly, data-driven approaches can also be used to predict motion based on raw IMU measurements collected by subjects who are free to carry their phone, which is difficult for PDR [63], [64]. We will introduce such data-driven approaches in our future work.

2) *The Accuracy of Regressing Motion Characteristics:* We use particle filtering technology to limit the drift of inertial navigation. Since particle filtering needs to fuse motion characteristics calculated by inertial sensors, its accuracy depends on the accuracy of distances and headings regressed by data-driven method. In this experiment, the subjects were asked to move around the experimental scene following three routes and collect IMU sensor data and Bluetooth RSS data in four usages under each route. The Bluetooth RSS data is collected to evaluate the performance of the fusion algorithm in the later experiments. Then we used the mean squared errors (MSE) as the metric to evaluate the accuracy of the proposed data-driven method in regressing distance and heading, as shown in Figure 9. For a more intuitive display, each node of the x-axis in Figure 9 used 400 frames (2 seconds) with the same size as the input window we selected. And the “distance” and “heading” of the y-axis indicate the corresponding increments. Assume that there is zero-distance and zero-heading regression values at the first sample of each route.

As shown, we observed that the errors of both DNN models are small. It can still be seen from the heading regression results that there are some occasional large errors in the heading estimation of IONet, while the smaller heading regression error in the proposed approach is a good verification of the superiority of introducing the coordinate frame alignment

module. Surprisingly, in most usages, the distance regression performance of the proposed method is superior to the method of IONet. For distance regression, our method achieved the mean squared errors of  $0.0079\text{m}^2$ ,  $0.0085\text{m}^2$ ,  $0.0063\text{m}^2$  and  $0.0081\text{m}^2$  on the texting, swinging, calling and pocket usage, respectively. And the mean squared errors of the heading regression for the four usages were  $0.0234\text{rad}^2$ ,  $0.0363\text{rad}^2$ ,  $0.0161\text{rad}^2$  and  $0.0152\text{rad}^2$ , respectively. Therefore, we can infer that the proposed approach is almost unaffected by the usages of smartphones, and can model the user’s movement well.

### B. Fusion Algorithm Performance

In this experiment, we utilized the data of the three fixed routes mentioned in the experiment above to implement particle filter localization. By analyzing the average localization errors of the three trajectories in all usages, we evaluated the proposed fusion algorithm.

1) *Effect of Particle Numbers:* We utilize particle filter to fuse inertial sensor-based localization and BLE-based localization for indoor positioning. As we all know, the number of particles directly affects the accuracy of particle filter localization. Indeed, the increase in the number of particles leads to an increase in calculation time. In order to make the fusion algorithm efficient and without losing its accuracy, we performed positioning with different numbers of particles on the three routes in texting usage. Using the data collected from the three routes, we can process in advance on the computer to obtain the BLE-based localization results and the results of displacements estimated by the two inertial sensor-based localization methods (the proposed method and the fusion method that utilizes particle filter to fuse traditional PDR and BLE-based localization). Next, by changing the number of particles and implementing particle filter fusion localization, we can evaluate the performance of the particle algorithm. The performance was evaluated in two metrics: localization error and computing time. Localization error is defined as the mean positional errors between the estimated position and the ground truth, and the computing time (taking the time obtained by implementing the fusion algorithm on a Macbook Pro in 2017 as a reference) is defined as the total time it takes to predict the three trajectories. For the case when the number of particles is 0, since the BLE-based localization results and the

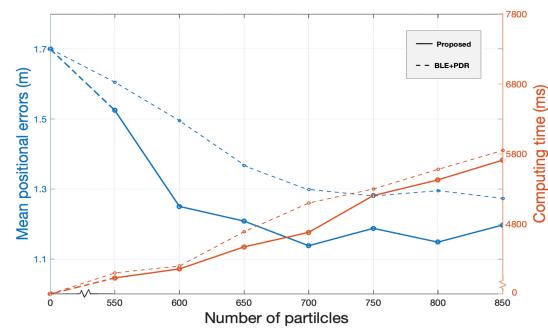


Figure 10: The effect of number of particles on the fusion algorithm performance

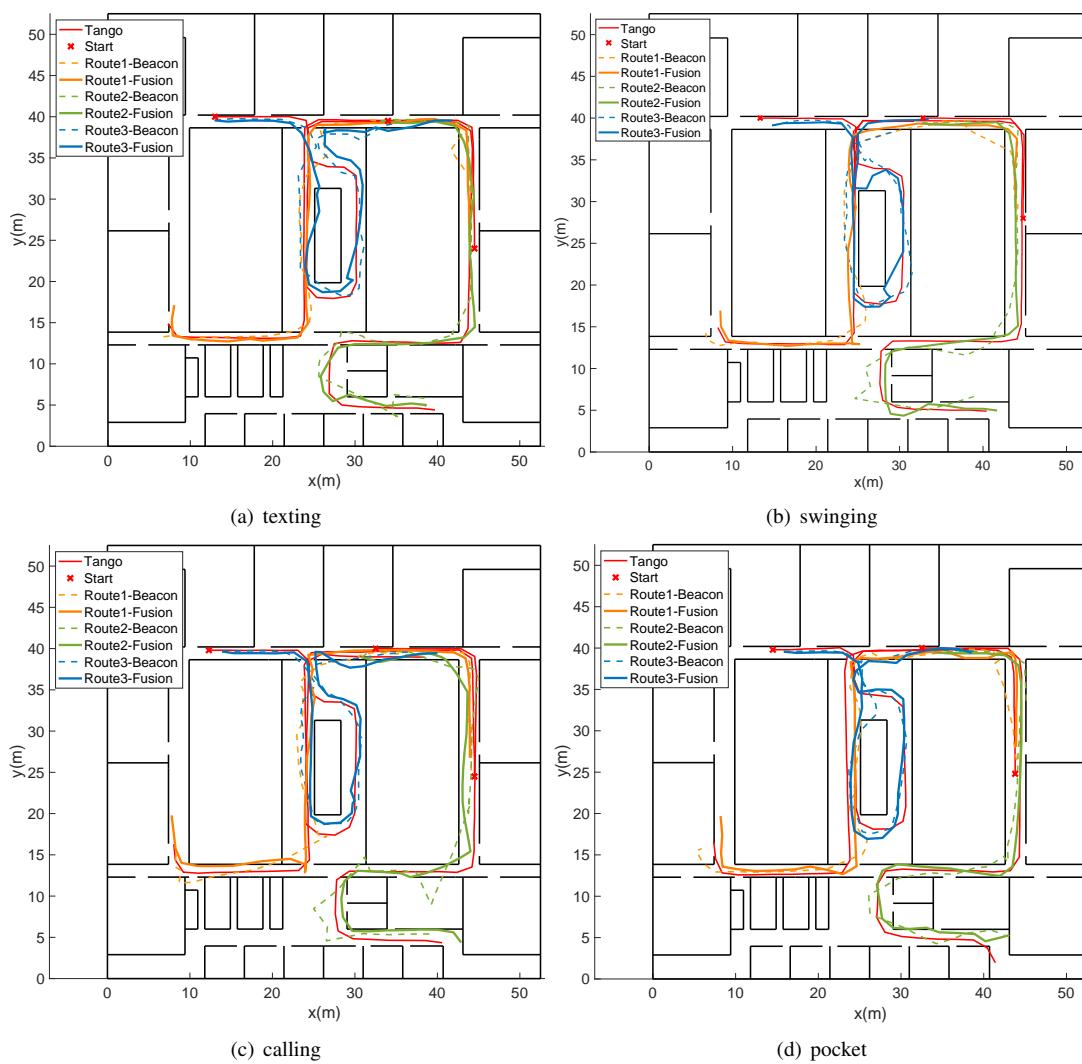


Figure 11: Comparison of trajectory prediction results between the proposed method and the BLE-based localization method

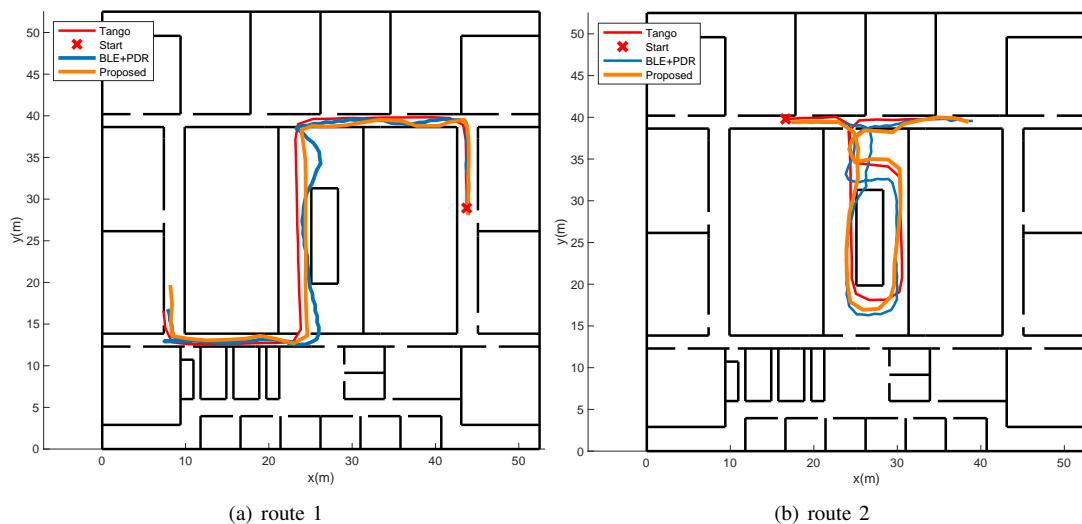


Figure 12: Comparison of trajectory prediction results between the proposed method and the existing fusion method under pocket usage

Table III: Positional accuracy evaluations

| Usages   | Methods  | MPE<br>(Route1) | MPE<br>(Route2) | MPE<br>(Route3) | MPE<br>(Mean) |
|----------|----------|-----------------|-----------------|-----------------|---------------|
| Texting  | BLE      | 1.7045          | 1.6886          | 1.7070          | 1.7000        |
|          | BLE+PDR  | 0.9454          | 1.4457          | 1.5046          | 1.2986        |
|          | Proposed | 0.7822          | 1.2164          | 1.4518          | 1.1501        |
| Swinging | BLE      | 2.0027          | 1.4808          | 1.4909          | 1.6581        |
|          | BLE+PDR  | 1.8002          | 1.3933          | 1.3730          | 1.5222        |
|          | Proposed | 1.6310          | 1.2001          | 1.1231          | 1.3181        |
| Calling  | BLE      | 1.6516          | 2.6134          | 1.2789          | 1.8480        |
|          | BLE+PDR  | 1.1233          | 2.5144          | 1.1000          | 1.5792        |
|          | Proposed | 1.1116          | 2.3146          | 0.9680          | 1.4647        |
| Pocket   | BLE      | 2.3706          | 1.4049          | 1.7513          | 1.8423        |
|          | BLE+PDR  | 1.5586          | 1.0964          | 1.5368          | 1.3973        |
|          | Proposed | 0.9610          | 1.1373          | 0.9212          | 1.0065        |

inertial sensor-based localization results are known in advance, we assume its computing time is 0 second. After evaluating the effect of different particle numbers on a computer, we can extend the optimal result to mobile phones. The different results of the two fusion methods are shown in Figure 10. As shown, we can see that when the number of particles is less than 700, the mean positional errors decrease as the number of particles increases. While the number is greater than 700, the errors tend to be stable. For the computing time, it gradually increases as the number of particles increases. Hence, we recommend setting the number of particles to 700.

2) *Localization Performance of Fusion Algorithm:* After fixing the number of particles, we implemented the proposed fusion algorithm and summarized the mean positional errors (in meters) of the three routes in Table III. We also compared the proposed method with BLE-based localization method and the fusion method that utilizes particle filter to fuse traditional PDR and BLE-based localization. As seen in Table III, the combination of triangulation method and fingerprinting method enables our BLE-based localization method to achieve a good mean positional error of 1.76m. Table III also validates the effectiveness of introducing particle filter to reduce localization errors, and the best accuracy of our proposed method.

On the one hand, compared with the BLE-based localization method, for the four usages (*i.e.*, *texting*, *swinging*, *calling* and *pocket*), the proposed fusion algorithm reduced the mean positional error by 32.35% (0.55m), 20.51% (0.34m), 20.74% (0.34m), and 45.37% (0.84m), respectively. Especially for route 1, the localization error of the proposed algorithm under *texting* usage is 0.78m, which is reduced by 54.12% over BLE (1.70m) and the error of the proposed algorithm under *pocket* usage is 0.96m, which is reduced by 59.49% over BLE (2.37m). These positioning results demonstrate that the proposed PF-based fusion algorithm can effectively improve the accuracy of indoor positioning system, which performs better than BLE-based method. More intuitively, the trajectory prediction results of the three routes in each usage are shown in Fig. 11. Clearly, the proposed approach, whose estimated trajectories match the ground truth closest, performs best in each usage test. Although the BLE-based localization results

have some occasional large errors (see Fig. 11), it provides useful localization information, which is contributed to the effective corrections of the motion characteristics regression errors of data-driven inertial navigation.

On the other hand, compared with the existing fusion method, the proposed fusion algorithm also further reduced the mean positional error of these four usages. Although the reduction is relatively small, and there is a case under *pocket* usage in route 2 where the accuracy of existing fusion method is better than that of our proposed algorithm, these cannot make us ignore the help of integrating data-driven method to improve the localization accuracy. For the mean positional error, the improvement results of route 1 and route 3 under *pocket* usage are more obvious, and we showed these trajectory prediction results in Fig. 12. The introduction of data-driven inertial navigation provides reliable help for constraining error drift of fusion system. By providing more accurate information about user movements, the prediction results of our proposed fusion method can be closer to the ground truth.

## V. CONCLUSION

In this paper, we proposed a feasible smartphone-based indoor localization algorithm by utilizing particle filter to leverage the fusion of data-driven inertial navigation and BLE-based localization. It is expected to benefit diverse smartphone applications that rely on a system component of data-driven inertial navigation. The experimental results show that the proposed PF-based fusion algorithm can effectively improve the accuracy of indoor positioning system, which performs better than BLE-based method and the fusion method that utilizes particle filtering to fuse traditional PDR and BLE-based localization. For data-driven inertial navigation, by adding the coordinate frame alignment module to optimize the IONet framework, we infer from the experimental results that compared with the PDR and IONet methods, the proposed method can better predict the distances and headings of pedestrian motion within 2 seconds in the four basic smartphone usages. In addition, the combination of triangulation method and fingerprinting method enables our BLE-based localization method to achieve a good mean positional error of 1.76m. Finally, in the case of a fixed number of particles of 700, we implemented a fusion positioning algorithm. On the one hand, compared with Bluetooth positioning, for the four usages of *texting*, *swinging*, *calling* and *pocket*, the proposed fusion algorithm reduced the mean positional error by 32.35% (0.55m), 20.51% (0.34m), 20.74% (0.38m), and 45.37% (0.84m), respectively. On the other hand, compared with the existing fusion method, the proposed fusion algorithm also reduced the mean positional error of these four usages, verifying its help for localization accuracy improvement. In future works, we will further expand the proposed approach to the situation when the placement of a smartphone is unconstrained.

## REFERENCES

- [1] B. Zhou, Q. Li, Q. Mao, W. Tu, and X. Zhang, “Activity sequence-based indoor pedestrian localization using smartphones,” *IEEE Transactions on Human-Machine Systems*, vol. 45, no. 5, pp. 562–574, 2017.

- [2] W. Ma, Q. Li, B. Zhou, W. Xue, and Z. Huang, "Location and 3d visual awareness-based dynamic texture updating for indoor 3d model," *IEEE Internet of Things Journal*, vol. PP, no. 99, pp. 1–1, 2020.
- [3] P. Bahl and V. N. Padmanabhan, "Radar: an in-building rf-based user location and tracking system," in *Infocom Nineteenth Joint Conference of the IEEE Computer & Communications Societies IEEE*, 2000.
- [4] W. Xue, K. Yu, Q. Li, B. Zhou, J. Zhu, Y. Chen, W. Qiu, X. Hua, W. Ma, and Z. Chen, "Eight-diagram based access point selection algorithm for indoor localization," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 13 196–13 205, 2020.
- [5] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil, "Landmarc: indoor location sensing using active rfid," in *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003. (PerCom 2003)*, 2003.
- [6] M. S. Barghi and R. De Groote, "Indoor localization based on response rate of bluetooth inquiries," in *Acm International Workshop on Mobile Entity Localization & Tracking in Gps-less Environments*, 2008.
- [7] B. Zhou, W. Ma, Q. Li, N. El-Sheimy, Q. Mao, Y. Li, F. Gu, L. Huang, and J. Zhu, "Crowdsourcing-based indoor mapping using smartphones: A survey," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 177, pp. 131–146, 2021.
- [8] B. Zhou, Q. Li, Q. Mao, W. Tu, X. Zhang, and L. Chen, "Alimc: Activity landmark-based indoor mapping via crowdsourcing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2774–2785, 2015.
- [9] O. J. Woodman and C. Woodman, "An introduction to inertial navigation," *Journal of Navigation*, vol. 9, no. 3, 2007.
- [10] P. Davidson and R. Pich, "A survey of selected indoor positioning methods for smartphones," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 1347–1370, 2017.
- [11] N. El-Sheimy, H. Hou, and X. Niu, "Analysis and modeling of inertial sensors using allan variance," *IEEE Transactions on Instrumentation & Measurement*, vol. 57, no. 1, pp. 140–149, 2007.
- [12] A. Solin, S. Cortes, E. Rahtu, and J. Kannala, "Pivo: Probabilistic inertial-visual odometry for occlusion-robust navigation," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2018.
- [13] T. Schneider, M. Dymczyk, M. Fehr, K. Egger, S. Lynen, I. Gilitschenski, and R. Siegwart, "Maplab: An open framework for research in visual-inertial mapping and localization," *IEEE Robotics & Automation Letters*, vol. 3, no. 3, pp. 1418–1425, 2018.
- [14] Harle and Robert, "A survey of indoor inertial positioning systems for pedestrians," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1281–1293, 2013.
- [15] C. Chen, X. Lu, A. Markham, and N. Trigoni, "Ionet: Learning to cure the curse of drift in inertial odometry," 2018.
- [16] S. Corts, A. Solin, and J. Kannala, "Deep learning based speed estimation for constraining strapdown inertial navigation on smartphones," 2018.
- [17] H. Yan, Q. Shan, and Y. Furukawa, "Ridi: Robust imu double integration," 2017.
- [18] M. Brossard, A. Barrau, and S. Bonnabel, "Ai-imu dead-reckoning," *IEEE Transactions on Intelligent Vehicles*, vol. 5, no. 4, pp. 585–595, 2020.
- [19] T. Haarnoja, A. Ajay, S. Levine, and P. Abbeel, "Backprop kf: Learning discriminative deterministic state estimators," 2016.
- [20] X. Zhao, Z. Xiao, A. Markham, N. Trigoni, and Y. Ren, "Does btle measure up against wifi? a comparison of indoor location performance," in *European Wireless 2014; 20th European Wireless Conference*, 2014, pp. 1–6.
- [21] Y. Wang, X. Yang, Y. Zhao, Y. Liu, and L. Cuthbert, "Bluetooth positioning using rssi and triangulation methods," in *Consumer Communications & Networking Conference*, 2013.
- [22] Q. D. Vo and P. De, "A survey of fingerprint-based outdoor localization," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 491–506, 2017.
- [23] B. Zhou, W. Tu, K. Mai, W. Xue, W. Ma, and Q. Li, "A novel access point placement method for wifi fingerprinting considering existing aps," *IEEE Wireless Communications Letters*, vol. 9, no. 11, pp. 1799–1802, 2020.
- [24] F. Ye, R. Chen, G. Guo, X. Peng, and L. Huang, "A low-cost single-anchor solution for indoor positioning using ble and inertial sensor data," *IEEE Access*, vol. PP, no. 99, pp. 1–1, 2019.
- [25] Z. Chen, Q. Zhu, and Y. C. Soh, "Smartphone inertial sensor-based indoor localization and tracking with ibeacon corrections," *IEEE Transactions on Industrial Informatics*, vol. 12, no. 4, pp. 1540–1549, 2016.
- [26] X. Liu, B. Zhou, P. Huang, W. Xue, Q. Li, J. Zhu, and L. Qiu, "Kalman filter-based data fusion of wi-fi rtt and pdr for indoor localization," *IEEE Sensors Journal*, vol. 21, no. 6, pp. 8479–8490, 2021.
- [27] Y. Yu, R. Chen, L. Chen, G. Guo, and Z. Liu, "A robust dead reckoning algorithm based on wi-fi ftn and multiple sensors," *Remote Sensing*, vol. 11, no. 5, pp. 504–, 2019.
- [28] L. Jose, Z. Zhao, T. Braun, and Z. Li, "A particle filter-based reinforcement learning approach for reliable wireless indoor positioning," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 99, pp. 2457–2473, 2019.
- [29] S. Maskell and N. Gordon, "A tutorial on particle filters for on-line nonlinear/non-gaussian bayesian tracking," in *Target Tracking: Algorithms & Applications*, 2001.
- [30] C. Wu, F. Zhang, B. Wang, and K. J. R. Liu, "Easitrack: Decimeter-level indoor tracking with graph-based particle filtering," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 2397–2411, 2020.
- [31] T. Dinh, N. S. Duong, and K. Sandrasegaran, "Smartphone-based indoor positioning using ble ibeacon and reliable lightweight fingerprint map," *IEEE Sensors Journal*, vol. PP, no. 99, pp. 1–1, 2020.
- [32] Xia, Hao, Zuo, Jinbo, Liu, Shuo, Qiao, and Yanyou, "Indoor localization on smartphones using built-in sensors and map constraints," *IEEE Transactions on Instrumentation and Measurement*, 2019.
- [33] Zheng, Yang, Chenshu, Wu, Zimu, Zhou, Xinglin, Zhang, Xu, and Wang, "Mobility increases localizability: A survey on wireless indoor localization using inertial sensors," *Acm Computing Surveys*, 2015.
- [34] Gong, Goldman, and Lach, "Deepmotion: a deep convolutional neural network on inertial body sensors for gait assessment in multiple sclerosis," in *Wireless Health*, 2016, pp. 1–8.
- [35] J. Hannink, T. Kautz, C. Pasluosta, J. Barth, S. Schülein, K.-G. Gaßmann, J. Klucken, and B. Eskofier, "Stride length estimation with deep learning," *ArXiv*, vol. abs/1609.03321, 2016.
- [36] M. Gadaleta and M. Rossi, "Idnet: Smartphone-based gait recognition with convolutional neural networks," *Pattern Recognition*, pp. 25–37, 2016.
- [37] A. Shrestha and M. Won, "Deepwalking: Enabling smartphone-based walking speed estimation using deep learning," 2018.
- [38] S. Y. Cho and C. G. Park, "Mems based pedestrian navigation system," *Journal of Navigation*, vol. 59, no. 1, pp. 135–153, 2005.
- [39] H. Weinberg, "Using the adxl202 in pedometer and personal navigation applications," *Analog Devices AN-602 application note*, vol. 2, no. 2, pp. 1–6, 2002.
- [40] J. Kim, H. Jang, D.-H. Hwang, and C. Park, "A step, stride and heading determination for the pedestrian navigation system," *Journal of Global Positioning Systems*, vol. 3, pp. 273–279, 12 2004.
- [41] W. Liu, D. Caruso, E. Ilg, J. Dong, A. I. Mourikis, K. Daniilidis, V. Kumar, and J. Engel, "Tlio: Tight learned inertial odometry," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5653–5660, 2020.
- [42] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.
- [43] R. Faragher and R. Harle, "An analysis of the accuracy of bluetooth low energy for indoor positioning applications," vol. 1, pp. 201–210, 01 2014.
- [44] C. P. Vicente, C. A. Anna, P. A. Josep, and P. B. Mara, "A bluetooth low energy indoor positioning system with channel diversity, weighted trilateration and kalman filtering," *Sensors*, vol. 17, no. 12, p. 2927, 2017.
- [45] P. Kriz, F. Maly, and T. Kozel, "Improving indoor localization using bluetooth low energy beacons," *Mobile Information Systems*, 2016(2016-4-19), vol. 2016, no. pt.2, pp. 1–11, 2016.
- [46] R. Faragher and R. Harle, "Location fingerprinting with bluetooth low energy beacons," *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 11, pp. 2418–2428, 2015.
- [47] J. Gomez-Gil, R. Ruiz-Gonzalez, S. Alonso-Garcia, and F. Gomez-Gil, "A kalman filter implementation for precision improvement in low-cost gps positioning of tractors," *Sensors*, vol. 13, no. 11, p. 1530715323, Nov 2013. [Online]. Available: <http://dx.doi.org/10.3390/s131115307>
- [48] L. Xin, W. Jian, and L. Chunyan, "A bluetooth/pdr integration algorithm for an indoor positioning system," *Sensors*, vol. 15, no. 10, pp. 24 862–24 885, 2015.
- [49] R. V. D. Merwe, E. A. Wan, and S. I. Julier, "The unscented kalman filter for nonlinear estimation," 2000, pp. 153–158.
- [50] A. Belmonte-Hernandez, G. Hernandez-Penalosa, D. M. Gutierrez, and F. Alvarez, "Swiblux: Multi-sensor deep learning fingerprint for precise real-time indoor tracking," *IEEE Sensors Journal*, vol. PP, no. 99, pp. 1–1, 2019.

- [51] D. Fox, W. Burgard, F. Dellaert, and S. Thrun, "Monte carlo localization: Efficient position estimation for mobile robots," in *Of the National Conference on Artificial Intelligence*, 1999.
- [52] M. Montemerlo, S. Thrun, D. Roller, and B. Wegbreit, "Fastslam 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, 2003, p. 11511156.
- [53] B. Zhou, J. Yang, and Q. Li, "Smartphone-based activity recognition for indoor localization using a convolutional neural network," *Sensors*, vol. 19, no. 3, 2019.
- [54] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Camera-imu-based localization: Observability analysis and consistency improvement," *The International Journal of Robotics Research*, vol. 33, no. 1, pp. 182–201, 2014.
- [55] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visualinertial odometry using nonlinear optimization," *International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2014.
- [56] K. Greff, R. K. Srivastava, J. Koutn, B. R. Steunebrink, and J. Schmidhuber, "Lstm: A search space odyssey," *IEEE Transactions on Neural Networks & Learning Systems*, vol. 28, no. 10, pp. 2222–2232, 2016.
- [57] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *Computerence*, 2014.
- [58] L. Shu, Y. Jiang, and A. Striegel, "Face-to-face proximity estimationusing bluetooth on smartphones," *IEEE Transactions on Mobile Computing*, vol. 13, no. 4, pp. 811–823, 2014.
- [59] S. Pradhan, Y. Bae, J. Y. Pyun, N. Y. Ko, and S. S. Hwang, "Hybrid toa trilateration algorithm based on line intersection and comparison approach of intersection distances," *Energies*, vol. 12, no. 9, p. 1668, 2019.
- [60] A. R. Jimenez, F. Seco, C. Prieto, and J. Guevara, "A comparison of pedestrian dead-reckoning algorithms using a low-cost mems imu," in *IEEE International Symposium on Intelligent Signal Processing*, 2009.
- [61] I. Constandache, R. R. Choudhury, and I. Rhee, "Towards mobile phone localization without war-driving," in *2010 Proceedings IEEE INFOCOM*, 2010, pp. 1–9.
- [62] J. A. B. Link, P. Smith, N. Viol, and K. Wehrle, "Footpath: Accurate map-based indoor navigation using smartphones," in *International Conference on Indoor Positioning & Indoor Navigation*, 2011.
- [63] H. Yan, S. Herath, and Y. Furukawa, "Ronin: Robust neural inertial navigation in the wild: Benchmark, evaluations, and new methods," 2019.
- [64] C. Chen, Y. Miao, C. X. Lu, L. Xie, and N. Trigoni, "Motiontransformer: Transferring neural inertial tracking between domains," in *The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-2019)*, 2019.



**Jianfan Chen** received the B.E. degree in electronic and information engineering from Shenzhen University, China, in 2018. He is currently pursuing the Ph.D. degree at Shenzhen University. His main research interests include the design and implementation of indoor tracking and navigation systems, deep learning algorithm design, and wireless communications.



**Baoding Zhou** received the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2015. He is currently an assistant professor with the College of Civil and Transportation Engineering, Shenzhen University, China. His research interests include indoor localization, indoor mapping, intelligent sensing, and intelligent transportation system.



**Shaoqian Bao** received the B.S. degree in Computer Science from Jiangxi Science & Technology Normal University, China, in 2016. He is currently working as an Algorithm Engineer at E-scope Spatial Intelligence Science Technology Co.,Ltd. His main research interests are the design and implementation of indoor localization and navigation systems, automation system engineering, and deep learning applications.



**Xu Liu** received the B.S. degree in geomatics engineering from Chengdu University of Technology, China, and the M.S. degree in geodesy and geomatics engineering from Kunming University of Science and Technology, China, in 2016 and 2019, respectively. He is currently a research assistant at Shenzhen University. His main research interests include the design and implementation of indoor tracking and navigation systems, deep learning algorithm design, UAV positioning, and image recognition.



**Zhining Gu** received the B.S. degree in geographic information science from Harbin Normal University, China, in 2019. He is currently pursuing the M.E. degree at Shenzhen University. His main research interests include the design and implementation of indoor tracking and navigation systems, deep learning algorithm design, and intelligent transportation.



**Linchao Li** received the Ph.D. degree from Southeast University in 2019. He was a visiting scholar with the University of Wisconsin-Madison from 2017 to 2018. He is currently an assistant professor at the Institute of Urban Smart Transportation & Safety Maintenance, Shenzhen University, China. His research interests include intelligent transportation system, machine learning, and signal processing.



**Yangping Zhao** received the Ph.D. degree in Electrical Engineering from Ecole Polytechnique de Montreal, Montreal, Canada, in 2018. He is currently an Associate Research Fellow at the College of Civil and Transportation Engineering of Shenzhen University, China. His research interests include IoT, Intelligent Sensing, Wireless Energy Harvesting, Structural Health Monitoring of Infrastructure.



**Jiasong Zhu** received the B.E. degree from the Huazhong University of Science and Technology, Wuhan, China, the M.E. degree from Wuhan University, Wuhan, and the Ph.D. degree from The University of Hong Kong, Hong Kong, in 1997, 2003, and 2008, respectively. He is currently an Professor with the College of Civil Engineering, Shenzhen University, Shenzhen, China. His research interests include multisensor integration and data fusion, high-resolution image processing, and GIS applications in urban planning and transportation.



**Qingquan Li** is the President of Shenzhen University. He received his Ph.D. in photogrammetry and remote sensing from Wuhan Technical University of Surveying and Mapping. For over twenty years, Dr. Li has been engaged in the developments of surveying engineering and their innovation applications, contributing especially to the principles and practices of dynamic precision engineering survey. His research interests include the integration of GNSS, RS and GIS, precision engineering survey, as well as spatiotemporal big data analytics. Dr. Li's work

has been recognized with a variety of distinctions. He is an Academician of IEAS, member of Committee at Science and Technology Commission of Ministry of Education, China, as well as chief-editor of the Journal of Geodesy and Geoinformation Science. His research was awarded National Scientific and Technological Progress Second Prize (China) and MoE Scientific and Technological Progress First Prize (China).