PeTrack: Smartphone-based Pedestrian Tracking in Underground Parking Lot

Xiaotong Ren, Shuli Zhu, Chuize Meng, Shan Jiang, Xuan Xiao, Dan Tao, and Ruipeng Gao*
Beijing Jiaotong University
Beijing 100044, China
{xiaotren, zhushuli, mengchuize, hilljiang, xiaoxuan, dtao, rpgao}@bjtu.edu.cn

2022 18th International Conference on Mobility, Sensing and Networking (MSN) | 978-1-6654-6457-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/MSN57253.2022.00122

***Abstract*—Although location awareness is prevalent outdoors due to GNSS systems and devices, pedestrians are back into darkness in indoor buildings such as underground parking lots. Frequently we forget where we park the car and get confused by such maze-like structure. In order to track pedestrians without any additional equipment and map support, we propose *PeTrack* which is a smartphone-only approach that collects the inertial measurement unit (IMU) data for long-term tracking. Our intuition is to train the tracking model with crowdsourced outdoor trajectories, and infer customized user's trace with only inertial readings at indoors. Specially, we propose an inertial sequence learning framework with outdoor geo-tags. We also exploit opportunistic landmark detection and structure cues to refine the trajectory. We have developed a prototype and conducted experiments in an underground parking lot, and results have shown our effectiveness.**

***Index Terms*—Inertial tracking, indoor localization, mobile crowdsensing**

I. Introduction

Thanks to the mature deployment of Global Navigation Satellite System (GNSS) such as GPS and BeiDou, people can easily acquire their locations at anytime outdoors, and such location awareness enables convenient navigation services in daily travelling. However, when we enter GNSS denied environments such as underground or multi-storey parking lots, satellite signals are difficult to penetrate, and location-based services may go unavailable. Therefore, pedestrians tracking without satellites to locate them and help them find the car parked when they return to the parking lot is essential. Fortunately, smartphone can capture the user's various data through multiple sensors (e.g., IMU, WIFI, Bluetooth, etc.). It is promising to achieve real-time location and navigation via extracting, fusing and learning of kinds of sensory readings.

In previous work, a straight forward method is to use Kalman filter [1], [2]. However, it causes unbounded tracking errors due to the accumulation by double integration. Another mainstream method is step counting [3]–[5], but it largely relies on a pre-defined stride length and holding posture, and accumulated errors also exist over long duration. Some other approaches [6], [7] fuse IMU and magnetometer for robustness, but the geomagnetic signals are always severely interfered by electromagnetic objects in underground parking lots. Furthermore, there are already several IMU datasets (e.g., RIDI [8], IoNet [9], RoNIN [10]) for indoor scenes, but most of them rely on accurate calibration and require additional equipment for ground truth collection, which is difficult for deployment on large scale.

In this paper, we propose *PeTrack* which is a smartphone-only pedestrian tracking approach in multi-level or underground parking lots. It can continuously track and record the pedestrian's walking trajectory from the drop-off point, so as to help them quickly find their car when they return to the parking lot.

However, such an inertial and phone-only solution entails a series of non-trivial challenges. First, it is difficult to represent pedestrian displacement through various mobile phone postures (e.g., hand-hold, pocket, calling). Second, IMU data is noisy, requiring robust modeling to decode the motion. Finally, long-term tracking will lead to the accumulation of errors. Therefore, the main contributions of this paper are as follows:

- We propose a data processing method based on the theory of rigid rotation, so as to transform smartphone's inertial data under arbitrary postures to the flat posture that is consistent with the pedestrian.
- We explore an inertial sequence learning framework that can predict the walking speed of pedestrians, via the Recurrent Neural Network(RNN) as the core algorithm and smartphone's IMU readings as the only input. We only need geo-tags to train the model, and infer indoor trajectories without any satellites or maps.
- We propose a trajectory refinement method by line-fitting, so as to eliminate the inertial drifts and accumulated errors. We also identify opportunistic landmark to further calibrate pedestrian's real-time location.

II. Related Work

Traditional Tracking Methods. The simplest and original method is to first use Kalman filter [1] to filter IMU data, and then integrate the processed acceleration to get the speed. After that, the displacement of pedestrians can be obtained [2]. However, this method will lead to amplification of data errors after twice integration, thus reducing the final effect. Another popular method is using the step detection [3]–[5] to get the walking distance of pedestrians. To get the walking direction

This work was supported in part by NSFC under Grant 62072029 and Grant 61872027, Beijing NSF Grant L192004, DiDi Research Collaboration Plan, and OPPO Research Fund. Ruipeng Gao is the corresponding author.

978-1-6654-6457-4/22/$31.00 ©2022 IEEE
DOI 10.1109/MSN57253.2022.00122

of pedestrians, You et al. [11] propose a hybrid method based on IMU and RSSI for pedestrian's dead reckoning. Tong et al. [12] use principal component analysis [13] to infer the direction of walking. However, compared with the vehicle always driving along the fixed lane, the walking pattern of the pedestrian is more uncertain, so the robustness of this method is poor.

Data-driven Prior Tracking Methods. RIDI is the first data-driven inertial navigation method [8], which focuses on correctly expressing the velocity vector in the device coordinate system while relying on multi-sensor fusion to estimate the heading. IONet [9] is a neural network-based method that regresses the velocity and the change rate of motion direction without relying on the direction information of external devices. They also provide a special dataset, which is collected in a specific room with Vicon equipment installed in advance. RoNIN [10] provides a large IMU dataset, which allows users to use mobile phones during the data collecting, and even sit down. However, its sampling frequency is as high as 200Hz, and the truth value needs to be collected by Google's Tango phone [14], which has been accurately calibrated and error compensated. The above methods require accurate calibration and additional equipment deployment to collect ground truth, so they are not suitable for large-scale deployment.

III. Design Overview

In order to track pedestrians at indoors, our intuition is to exploit outdoor trajectories with geo-tags to train the tracking model, and then apply it to infer indoor trajectories in real time.

The architecture of our *PeTrack* is shown in Fig. 1. It uses inertial measurements as the only input, and geo-tags as the groundtruth for training. Specially, *PeTrack* consists of three steps.

Step 1: Data processing. In order to synchronize smartphone's inertial readings and its geo-tags by GPS, we identify all turns during walking and calculate the average time difference between the two sensing modalities. After that, we used the game rotation vector to convert the inertial data from arbitrary posture to flat posture, whose coordinate system is consistent as the pedestrian. Finally, we devise a one-dimensional convolution operation for denoising.

Step 2: inertial sequence learning. We explore an LSTM [15] based model to learn the current walking speed directly from the inertial readings, rather than directly performing continuous integration or step counting. At indoors, we use the trained model to infer the pedestrian's walking speed without geo-tags.

Step 3: Trajectory refinement. We propose a line-fitting algorithm which corrects the orientation drifts for heading estimation during walking. We also identify multiple opportunistic landmarks such as the elevator, stairs, and entrances, so as to calibrate pedestrian's realtime locations in the parking lot.

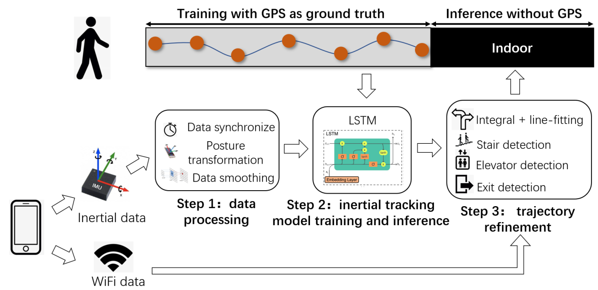


Fig. 1. Design overview.

IV. Methodology

A. Data Processing

Data synchronization. In the outdoor scenario, the GPS data is collected as the groundtruth. However, through observation, we find that there is always a 0-5s delay in GPS for the heading change of pedestrians, that is, when gyroscope has sensed the bearing change of pedestrians, GPS data may reflect the change after 5s at the latest. Therefore, IMU data needs to be aligned with GPS data.

Multi-pose transformation. We stipulate that under ideal conditions, when data is collected, the posture of smartphone should be completely flat, that is, the positive direction of mobile phone's Y axis is parallel to the horizontal plane and points to the forward direction of pedestrian. However, in the real world, the posture of the smartphone can be varied. For example, the user may talk on the phone while walking, or put the phone in his pocket. Therefore, it is necessary to rotate the collected inertial measurement unit (IMU) data from arbitrary posture to flat posture.

We use the game rotation vector to implement this process. As shown in Fig. 2, first of all, when the sensor is registered, the initial coordinate system of the mobile phone will be determined, and the IMU data in this coordinate system is set as $\boldsymbol{d_0}$. Set the value of game rotation vector at flat posture as $\boldsymbol{r_1}$ and IMU as $\boldsymbol{d_1}$. Set the value of game rotation vector at any attitude as $\boldsymbol{r_2}$ and the IMU data as $\boldsymbol{d_2}$. We can obtain 1 from the rotation formula of quaternions.

$$\boldsymbol{r_1^{-1}} \times \boldsymbol{r_2} \times \boldsymbol{d_2} \times \boldsymbol{r_2^{-1}} \times \boldsymbol{r_1} = \boldsymbol{d_1} \tag{1}$$

During the calibration process before the formal start of each track, we will let the pedestrian hold the hand-held flat posture for a period of time, so that r_1 is known, and the real-time r_2 and d_2 can be obtained when the pedestrian changes to any posture. Therefore, the IMU data can be rotated from arbitrary posture to flat posture by using 1.

Data smoothing. Since the current inertial sensors on smart phones do not have high precision, there is a large noise in the collected data. We use one-dimensional convolution to smooth and denoise the data. As shown in the Fig. 3, the outliers (values that are too large) are significantly smaller after smoothing, which is more conducive to model training.

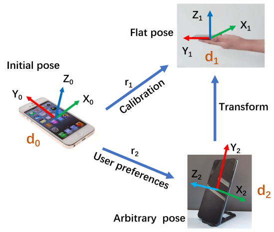


Fig. 2. Multi-pose transformation.

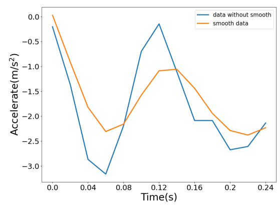


Fig. 3. Effect of data smoothing.

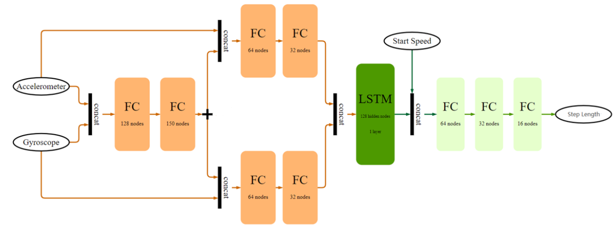


Fig. 5. Network structure of speed estimation model.

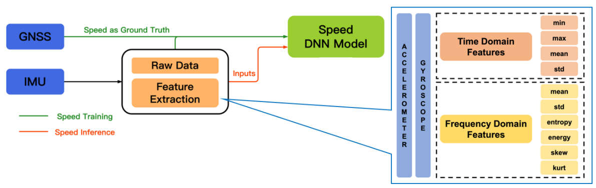


Fig. 4. Data flow of speed estimation.

B. Inertial Tracking Model Overview

We design a supervised deep learning model (Speed DNN Model) and decoupled the pedestrian speed estimation function from bearing estimation. The data flow of the speed estimation model is shown in Fig. 4. The whole workflow is divided into two lines: model training and model inference. In model training, we use inertial data collected outdoors to construct the training dataset with the speed provided by high-quality GNSS as the groundtruth. For inertial data processing, we design a feature extraction module to extract the feature of the accelerometer and gyroscope, including time domain feature and frequency domain feature.

C. Network Structure

The structure of speed estimation model is shwon in Fig. 5. The model is generally divided into three parts, among which the first part is embedding layer, the middle part is representation layer, and the last is regression layer.

Embedding layer. The embedded layer takes the features of accelerometer (ACC) and gyroscope (GYR) as input to realize the re-extraction and effective fusion of such inertial features. We first adopted a two-layer fully connected network to extract the fusion features of accelerometer and gyroscope. Then, a symmetric two-layer fully connected network is adopted, and the extracted feature information is fused with the accelerometer feature and gyroscope feature respectively.

The representation layer. For the design of the representation layer, we directly choose LSTM, a representative member of the recurrent neural network. The structure is designed with stack layer number of 1 and hidden layer dimension of 128.

The regression layer. The regression layer takes the output of the representation layer and the initial walking speed of pedestrians (Spd_{start}) as input, and designs a fully connected network with a depth of 4 layers. The output is the change of walking speed in this period of time (one second) compared to the initial speed.

The way of obtaining Spd_{start} is different in model training and model inference. As the training set of the model adopts outdoor data with GNSS information, the pedestrian speed provided by GNSS is directly adopted in the training Spd_{start}. However, GNSS cannot work when model reasoning is applied in indoor places. In order to fill the vacancy of Spd_{start}, the median of the average pedestrian speed in the first three seconds of each track in the data set (0.81m/s) is adopted as the default value of Spd_{start}.

Hyper-parameter. In the model training process, we use SmoothL1Loss function and Adam optimizer. The epochs are 80 and the initial learning rate is 0.001, decreasing by 50% every 10 rounds of training.

D. Trajectory Refinement

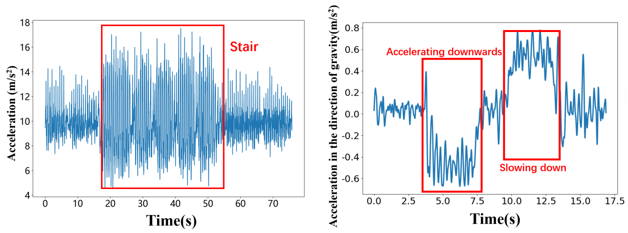
Heading Estimation. In order to estimation of the user's walking path, it is not enough to have the speed feature alone, but also need to obtain the user's heading information, *PeTrack* only uses the data from the accelerometer and gyroscope to estimate the bearing. We first integrate gyro data directly, because after rotation in data processing, the IMU data have been rotated to the ideal posture, so the z axis gyroscope data can be integrated directly without any further processing. Next, we combine the integration results with velocity estimation results to obtain the user's trajectory coordinates, and then use line-fitting algorithm to correct the trajectory coordinates.

Line-fitting algorithm. As we all know, the change in heading is more pronounced at the corner than in a straight line. We assume that the turning angles are all 90 degrees. When turning, the bearing changes gradually, so we consider the position where the bearing changes more than 45 degrees in 3s as the corner. We segment the trajectory according to the bearing, and use the least square method to fit the segmented trajectory.The bearing difference between the segmented trajectories is calculated according to the slope of the fitted line, and then the required correction angle is calculated by making the difference with 90 degrees. Finally, to obtain the new coordinate of the trajectory point, we rotate the trajectory around the starting point of the trajectory.

E. Landmark Detection

Landmark detection is mainly divided into two parts, stair and elevator detection, and opportunistic WiFi-based detection.

Stairs and Elevator Detection. As Fig. 6(a) shown, usually when pedestrian climbs stairs, the acceleration will change



(a) Stair detection (b) Elevator detection

Fig. 6. Acceleration features of stairs and elevators.

TABLE I
TRAJECTORY DATASET DETAILS

Scenarios	Trajectories	Duration	Frequency	Posture
outdoor	125	3-5min	50hz	flat
	74			calling
	74			pocket
indoor	50	2-5min	50hz	flat
	24			calling
	24			pocket

regularly. And as Fig. 6(b) shown, when the pedestrian is in the running elevator, the acceleration data also has obvious feature, so that we can use this feature to detect them. Specifically, we achieve a feature matching method by calculating the amplitude of acceleration and gait detection.

Opportunistic Wifi-based landmark detection. Although there are abundant WiFi signals in major shopping malls and office buildings, signals in underground parking lots are always weak or absent. However, sometimes there may be a relatively strong signal at the entrances of a parking lot, which are landmarks in this paper. Therefore, an opportunistic Wifi-based landmark detection can be conducted. In the matching algorithm, the Euclidean distance between the sample and the ground truth of the fingerprint database is calculated. When the Euclidean distance obtained is less than a certain threshold, the user can be judged to be at the landmark.

V. EVALUATION

A. Methodology

In order to evaluate the effect of the *PeTrack*, we collect data in both outdoor and indoor scenarios. The dataset details are listed in Table I. In order to simulate various postures of the mobile phone, we design three poses as shown in the Fig7: holding flat, making a phone call, and placing the mobile phone in the pants pocket. Data is collected for each posture indoors and outdoors respectively.

In the implementation, we use PyTorch to train our model on the machine learning platform with one GeForce GTX 2080 Ti GPU, an Intel i7 CPU, and 32G RAM.

B. Evaluation of Trajectory Tracking

Accuracy. We compare the *PeTrack* and classical step counting algorithm with direct gyroscope integration to obtain the heading. As shown in Fig. 8, the CDF curve represents the final point distance error of 50 flat posture tracks. The error

(a) Flat (b) Calling (c) Pocket

Fig. 7. Schematic diagram of smartphones with different postures.

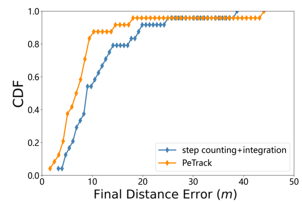


Fig. 8. Localization errors.

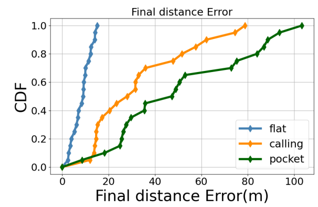


Fig. 9. Influences of three postures.

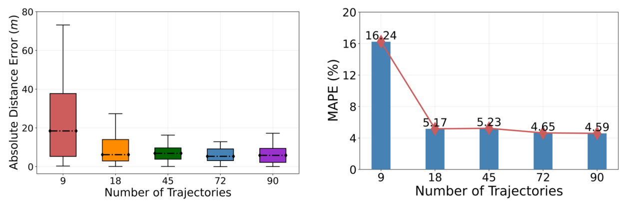
of *PeTrack* is less than 10m in 80% of cases, while the error of step counting is more than 15m.

Various posture. We also compare the performance of *PeTrack* on different posture, as shown in the Fig. 9. It can be seen that the performance of the flat posture is obviously better than the others. Especially, the pocket posture error is very large, which is mainly because when the mobile phone is placed in the pants pocket, pedestrians will regularly lift and lower their legs at every step when walking, which leads to a huge error in IMU data.

Training set size. We compare the effects of different training sets on model stability. Ninety walking tracks of flat posture (the walking duration of a single track is 3-5 minutes) are regarded as ”1” as a whole, and five grades are set according to percentage: 10%, 20%, 50%, 80% and 100%. The corresponding number of tracks are 9, 18, 45, 72 and 90 in sequence. The results are shown in Fig. 10(a), in which it can be found that when only 9 track data are used for training, the median error of the speed model is close to 20m. When the order of magnitude is gradually increased, the error range and median of the velocity model tend to be stable gradually, and the model achieves the best effect when 72 track data are used for training. Moreover, to make a fairer comparison, we introduced Mean Absolute Pencentage Error (MAPE), and the evaluation results are shown in Fig. 10(b). It can be found that MAPE gradually decreases with the increase of magnitude, which is consistent with the result in the figure above.

C. Landmark Detection Evaluation

Stair and elevator detection. We perform 2-minute walks on the ground and stairs, and collect IMU data for different postures during walking. For elevator detection, a total of 30 sets of data are collected, including elevator ascending and elevator descending categories. As shown in Table II, for all postures, the average accuracy of stair detection is about 96.1%, and the accuracy of elevator detection is about 99.0%. Because the feature is very obvious, and not affected



(a) (b)

Fig. 10. Error of speed model under different amount of training data.

TABLE II
STAIR/ELEVATOR DETECTION

Scenario	Recall	Accuracy
stair	100.0%	96.1%
elevator	100.0%	99.0%

TABLE III
WIFI-BASED DETECTION

Recall	Accuracy	Precision
89.8%	92.3%	87.8%

by the posture of the mobile phone, the detection difficulty is relatively low and the result is relatively good.

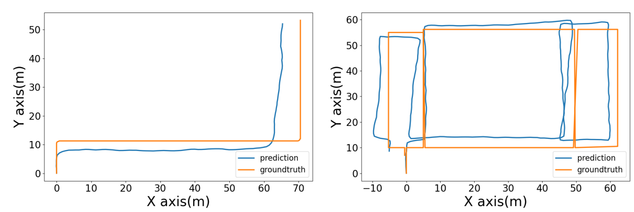
Opportunistic WiFi-based detection. The test results are shown in Table III. Because the WiFi signal in our experimental scenario is very weak, the recall rate is only 89.8%. Correspondingly, weak WiFi signals are occasionally detected in non-landmark locations, so misidentification also occurs, resulting in an precision rate of only 87.8%, as we continuously detect signals throughout the pedestrian's walk. This further proves the low applicability of WiFi location in indoor parking lots, and we only use it opportunistically as an auxiliary means.

D. Trajectory Generation

We further conducted experiments on the generation of pedestrian walking tracks. In the interior, a simple route and a complex route are selected as examples. The result is shown in Fig. 11. The simple route has a total length of about 120m and contains two turns, while the complex route has a total length of about 410m and contains 14 turning points. We carefully measured the distance of the planned paths to the starting point(the origin in Fig. 11), and plotted the groundtruth trajectories. It can be seen that no matter it is a simple trajectory or a complex trajectory containing multiple turns, our method can restore the trajectory well. For the simple trajectory, the final point position error is 6.14m, the heading error is 2.98°, and the position error of the two turning points is within 5m. For the complex trajectory, the final point position error is 6.36m, the heading error is 14.12°, and the position error of all turning points is also within 7m. Therefore, *PeTrack* can provide relatively accurate indoor pedestrian tracking and trajectory restoration.

VI. CONCLUSION

In this paper, we propose *PeTrack*, a smartphone-based indoor pedestrian tracking method that solves the problem without additional device deployment. The basic idea we follow is to train the model with outdoor data and apply the model to the indoor scene, so as to solve the problem that the model cannot be trained in the indoor scene due to the lack of groundtruth. By using *PeTrack*, even the data



(a) Simple route (b) Complex route

Fig. 11. Indoor trajectory generation.

for arbitrary phone posture can be converted to the flat pose for use. However, for the case that the original data itself has a large data drift, we cannot give a good tracking result temporarily, which is the main focus of our future work.

REFERENCES

[1] T. Basar, *A New Approach to Linear Filtering and Prediction Problems*, 2001, pp. 167–179.

[2] F. Aghili and C.-Y. Su, "Robust relative navigation by integration of icp and adaptive kalman filter using laser scanner and imu," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 4, pp. 2015–2026, 2016.

[3] W. Kang and Y. Han, "Smartpdr: Smartphone-based pedestrian dead reckoning for indoor localization," *IEEE Sensors Journal*, vol. 15, no. 5, pp. 2906–2916, 2015.

[4] H. Ngoc-Huynh, T. Phuc Huu, and G.-M. Jeong, "Step-detection and adaptive step-length estimation for pedestrian dead-reckoning at various walking speeds using a smartphone," *Sensors*, vol. 16, no. 9.

[5] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: Unsupervised indoor localization." New York, NY, USA: Association for Computing Machinery, 2012.

[6] Y. Shu, K. G. Shin, T. He, and J. Chen, "Last-mile navigation using smartphones." New York, NY, USA: Association for Computing Machinery, 2015.

[7] S. Shen, M. Gowda, and R. Roy Choudhury, "Closing the gaps in inertial motion tracking." New York, NY, USA: Association for Computing Machinery, 2018.

[8] H. Yan, Q. Shan, and Y. Furukawa, "Ridi: Robust imu double integration," in *15th European Conference on Computer Vision (ECCV)*, ser. Lecture Notes in Computer Science, vol. 11217, pp. 641–656.

[9] C. Chen, X. Lu, A. Markham, N. Trigoni, and Aaai, "Ionet: Learning to cure the curse of drift in inertial odometry," in *32nd AAAI Conference on Artificial Intelligence / 30th Innovative Applications of Artificial Intelligence Conference / 8th AAAI Symposium on Educational Advances in Artificial Intelligence*, ser. AAAI Conference on Artificial Intelligence, pp. 6468–6476.

[10] S. Herath, H. Yan, Y. Furukawa, and Ieee, "Ronin: Robust neural inertial navigation in the wild: Benchmark, evaluations, new methods," in *IEEE International Conference on Robotics and Automation (ICRA)*, ser. IEEE International Conference on Robotics and Automation ICRA, pp. 3146–3152.

[11] Y. You and C. Wu, "Hybrid indoor positioning system for pedestrians with swinging arms based on smartphone imu and rssi of ble," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–15, 2021.

[12] Y. Tong, S. Zhu, X. Ren, Q. Zhong, D. Tao, C. Li, L. Liu, and R. Gao, "Vehicle inertial tracking via mobile crowdsensing: Experience and enhancement," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–13, 2022.

[13] J. Janardhanan, G. Dutta, and V. Tripuraneni, "Attitude estimation for pedestrian navigation using low cost mems accelerometer in mobile applications, and processing methods, apparatus and systems," Patent 20 160 290 807, October, 2016.

[14] "Google tango developer guide," https://developers.google.com/tango/overview/motiontracking.

[15] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 11 1997. [Online]. Available: https://doi.org/10.1162/neco.1997.9.8.1735