Indoor Localization by Kalman Filter based Combining of UWB-Positioning and PDR

Gang Toe Lee, Seung Beom Seo, and Wha Sook Jeon
Department of Computer Science and Engineering
Seoul National University, Korea
Email: gtlee@mccl.snu.ac.kr, sbseo@mccl.snu.ac.kr, wsjeon@snu.ac.kr

2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC) | 978-1-7281-9794-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/CCNC49032.2021.9369588

***Abstract*—Ultra-wideband (UWB) ranging-based indoor positioning system (IPS) is suggested as one of the solutions to satisfy the requirement of decimeter level accuracy in indoor environments. However, UWB ranging error caused in non-line-of-sight (NLOS) environments is inevitable, which reduces accuracy of positioning. To tackle this problem, in this paper, an indoor positioning scheme that combines UWB positioning with pedestrian dead reckoning (PDR) is designed and proposed. First, the proposed scheme improves the performance of PDR utilizing UWB positioning in order to achieve the effect of parameter adaptation used in PDR. To this end, step detection and stride length estimation in traditional PDR are substituted with a deep learning-based speed estimation. In addition, heading estimation is improved by calibrating tilt effect of smartphone with the aid of UWB positioning. Then, with the UWB-assisted PDR (U-PDR), we also propose UWB positioning and U-PDR fusion algorithm using Kalman filter (KF). The proposed fusion algorithm complements UWB positioning and U-PDR based positioning, which improves the accuracy of positioning. Experimental results demonstrates that the performance of the proposed algorithm is better than that of UWB positioning or PDR only algorithms.**

***Index Terms*—indoor positioning system (IPS), ultra wide band (UWB) positioning, pedestrian dead reckoning (PDR), inertial sensor measurement unit (IMU), deep learning, long short term memory (LSTM), Kalman filter (KF)**

I. Introduction

Recently, indoor localization has attracted great attention from both the academic and industrial communities. For the location-based services (LBSs) such as indoor navigation and smart museum guide [1] – [2], accurate location of the target user (or device) is essential. Global positioning system (GPS) as one of the representative localization systems provides an accurate localization service in outdoor environments. According to [3], however, ordinary people spend most of their time indoors where GPS does not work properly because of signal attenuation [4]. Therefore, various indoor positioning systems (IPSs) have been proposed and widely used, which can be categorized into two groups: infrastructure-based and infrastructure-free systems.

WiFi, bluetooth low energy (BLE), and ultra-wideband (UWB) are representative radio frequency (RF) systems used for indoor positioning, which require infrastructure such as access point (AP) of WiFi, beacon device of BLE, and anchor node of UWB. Among them, UWB can achieve highly accurate ranging (i.e., estimating distance between two devices) due to the wide bandwidth and short transmission duration of signal. In order to estimate the position of a user, ranging values between the tag node held by or attached in the user and at least three anchor nodes are needed. Since the coordinates of the anchor nodes are known in advance, the position of the target user can be calculated through well-known multilateration algorithm. Thus, the accuracy of UWB-based indoor positioning highly depends on the accuracy of ranging. However, in case that one or more ranging values are measured in non-line-of-sight (NLOS) environments, ranging errors may be occurred which incur positioning error.

On the other hand, pedestrian dead reckoning (PDR) which is one of the well-known infrastructure-free IPS is relatively less affected by environments. PDR utilizes the patterns of pedestrian and it is implemented with inertial measurement unit (IMU) sensors composed of accelerometer, gyroscope, and magnetometer. Step detection, stride length estimation, and heading estimation are the main functions of PDR, by which relative movements can be calculated. In other words, PDR calculates temporary movement of a user by using the IMU sensors and estimates the current position of the user by recursively adding the instantaneous movement to the previous position. Although PDR can estimate relatively accurate position for short-time intervals, it suffers from the cumulative errors of IMU sensors as time goes by [5]. Furthermore, since the accuracy of PDR depends on the precision of the previous position as well as that of IMU sensors, the performance of PDR can be improved by other assistant methods.

In [6], received signal strength indicators (RSSIs) from WiFi APs are used to compensate PDR drift. In [7], [8], and [9], the authors utilize ultrasound, near field communication (NFC) tags, and radio frequency identification (RFID) to assist PDR. Ultrasound beacons, NFC tags, and RFID readers were used as landmarks because their coordinates are already known. The RSSI of BLE also used to estimate distance between two BLE devices using a path-loss model in [10]. Similarly to the above researches, it can correct the cumulative error of IMU-based PDR. Authors in [11] integrated UWB and PDR for indoor positioning, but since the IMU sensor was mounted on the foot of the pedestrians, there is a limitation in general use

This work was supported by the National Research Foundation of Korea (NRF) through the Ministry of Science and ICT (MSIT) of Korea Government under Grant NRF-2020R1F1A1075882.

Authorized licensed use limited to: ULAKBIM UASL - Galatasaray Universitesi. Downloaded on December 23,2024 at 21:46:19 UTC from IEEE Xplore. Restrictions apply.

cases.

The fusion methods in [6] – [11] show better performance comparing to the schemes using PDR only. However, the traditional PDR schemes have fundamental shortcoming in which the control parameters of stride length estimation cannot be adaptively applied to each person individually. That is, it is difficult to be widely used for many persons with a fixed control parameters. Thus, in this paper, we propose UWB-assisted PDR (U-PDR) utilizing deep learning approach, which achieves relative high accuracy for many people by adapting related parameters to each person automatically. Specifically, U-PDR learns personal patterns that appear when pedestrians walk and calibrates heading using UWB-based positioning results. In addition, step detection and stride length estimation steps are replaced with a speed estimation in the proposed U-PDR, which results in the more accurately estimated position. The speed estimation is based on deep learning where the measurements from IMU sensors and the speed of a pedestrian are input and output, respectively. Then, we finally propose Kalman filter (KF) based positioning algorithm where result of the UWB-based positioning (i.e., multilateration) and that of U-PDR are fused as measurements and estimates of KF, respectively. By combining the UWB and U-PDR, the weak points of UWB-based positioning and U-PDR, i.e., errors from UWB in NLOS environments and cumulative errors in U-PDR, can be mitigated. The experimental results show that the performance of the proposed algorithm is better than that in other schemes using UWB alone or PDR only.

The rest of the paper is organized as follows. The next section describes the proposed UWB/U-PDR combined indoor positioning algorithm, which focus on U-PDR and KF-based UWB/U-PDR fusion algorithm. In Section III, several experiments for each algorithm are explained and the results are evaluated. Finally, Section IV concludes this paper.

II. UWB/U-PDR COMBINED INDOOR POSITIONING ALGORITHM

Both UWB positioning and U-PDR are kinds of IPSs, but they have quite different characteristics. The feasible interval of UWB positioning is longer than that of measurement from IMU sensors, That is, multiple results from U-PDR (i.e., estimated speed and heading) can be obtained while UWB positioning estimates a position. In addition, in NLOS environment, the accuracy of UWB positioning is decreased because of the ranging error. However, in the early time, U-PDR may produce quite accurate estimations while the error may be accumulated and the accuracy is decreased as the pedestrian walks in long time.

Based on the characteristics of UWB positioning and U-PDR, their pros and cons can complement each other using KF. KF is already known as good for position tracking, because it uses a series of measurements observed over time, containing noise and other inaccuracies, and produces estimates of unknown variables that tend to be more accurate than those based on a single measurement alone. By adopting the result of UWB positioning as the measurements and the result of U-PDR as estimations of KF, respectively, the results from UWB positioning and U-PDR can complement each other to offset the disadvantages and highlight the advantages.

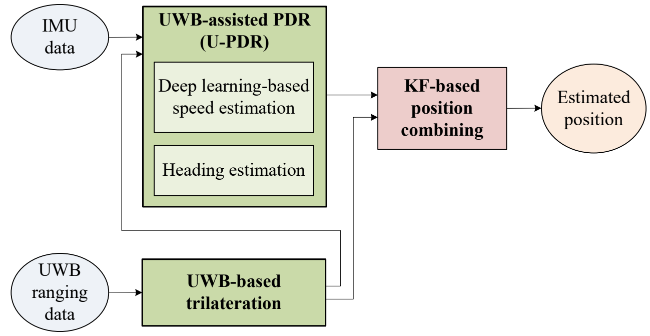

Fig. 1. System overview

The first part of the proposed algorithm is U-PDR which calibrates PDR using results of UWB positioning to reduce the accumulative error. In the traditional PDR, estimating traveled distance and heading is needed to get the current position of the user from the previous position. Unlike the traditional ways, however, in our approach we estimate the speed of the user to get distance and we also estimate the heading (i.e., traveled direction) assuming that a pedestrian walks straight during short time interval. Based on our approach, the proposed U-PDR is divided into two parts, which are 1) speed estimation and 2) heading estimation.

For speed estimation, to improve accuracy of the speed estimation, a deep learning model is implemented to adapt the personal patterns of each pedestrian. The model trained in the offline phase may estimate the speed of the target user based on the real-time measurements from IMU sensors measured in the online phase. In the proposed U-PDR, several parameters related heading estimation can be calibrated when the pedestrian walks straight enough to identify the heading of the user by UWB positioning. In practice, our method is helpful for estimating the actual heading of the user because pedestrians walk straight ahead in many cases.

The second part of the proposed algorithm is UWB/U-PDR combined KF positioning algorithm, to take the advantages of UWB positioning and U-PDR. The proposed algorithm combines the advantage of UWB positioning and U-PDR by using KF with UWB positioning results as measurements and PDR positioning results as estimates. Fig. 1 shows the overview of the proposed positioning algorithm. The algorithm will be described in detail in the following subsections.

A. UWB-assisted PDR (U-PDR)

In PDR, the current position is represented by the following equation.

$$\begin{pmatrix} X_t \\ Y_t \end{pmatrix} = \begin{pmatrix} X_{t-1} \\ Y_{t-1} \end{pmatrix} + \begin{pmatrix} l_t \cos\theta_t \\ l_t \sin\theta_t \end{pmatrix} \tag{1}$$

where X_t and Y_t are spatial coordinates at time t while l_t and θ_t are stride length and heading direction, respectively. Gener-

Authorized licensed use limited to: ULAKBIM UASL - Galatasaray Universitesi. Downloaded on December 23,2024 at 21:46:19 UTC from IEEE Xplore. Restrictions apply.

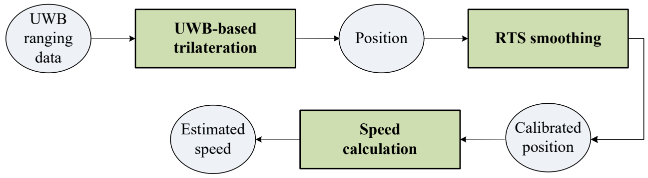

Fig. 2. The transformation from UWB ranging to speed

ally, in legacy PDR the accelerometer is used to estimate the stride length and the number of steps whereas the gyroscope and the magnetometer are used to estimate the heading.

Unlike legacy PDR, in the proposed U-PDR, the current position (X_t, X_t) is obtained by estimating the speed (S_t) during short time interval (Δt), i.e., $l_t = S_t \Delta t$. That is, when Δt is given or constant value, the speed corresponds to the stride length.

1) Speed estimation: When people walk at various speeds, the distinctive patterns appear on IMU data [12]. Conventionally, in the legacy PDR schemes, parameters for these patterns are tuned manually for each pedestrian. However, by pattern training through the deep learning method, a personalized speed estimation model can be made without any extra efforts. To learn the distinctive patterns of user speeds, it is necessary to know the exact speed of the user during the pattern training, which can be provided from UWB positioning. If we calculate the exact speed with UWB positioning, the deep learning model can learn the speed pattern of the user by matching IMU raw data with user speed as input and output of the deep learning model, respectively.

The input of the learning model consists of 3-axis accelerometer, 3-axis gyroscope, and 3-axis magnetometer values. The estimation interval should be long enough to contain a meaningful pattern. Thus, the size of input is $R \times 9$ where R is the sampling rate of IMU sensors. For example, when IMU sensors are measured at 50 Hz and the interval of speed estimation is 1 second, so the size of the input data is 450 per every second.

To get the more accurate speed values from UWB positioning, a transformation of the UWB positioning result is needed. The transformation procedure is illustrated in Fig. 2. First, the position is calculated from UWB ranging values by multilateration, and the position is calibrated by Rauch-Tung-Striebel (RTS) smoothing. We can derive the accurate speed by calculating Euclidean distance between the current position and the previous position for unit time.

On the other hand, we choose long short-term memory (LSTM) model as the learning model in order to learn the correlation between measurements of IMU sensors and UWB-based speed values. This is because measurements of IMU sensors are time-series data and LSTM is known to show good performance for time-series data analysis. We compose the LSTM model as one input layer, three hidden layers, and one output layer. In the offline phase, the model is trained by using the aforementioned input and output data. After the training is finished, the real-time speed of the pedestrian is estimated

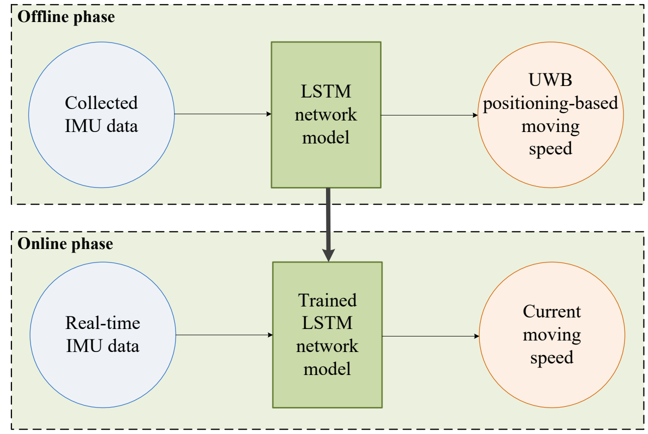

Fig. 3. Deep learning-based speed estimation

by the trained model using the real-time measurements from IMU sensors in the online phase. The overall process of deep learning-based speed estimation can be summarized as Fig. 3.

2) Heading estimation: In general, the heading of a device equipped with IMU sensors such as smartphone can be calculated based on magnetometer and gyroscope. In PDR with smartphones, it is assumed that the heading of a pedestrian is the same as the heading of the smartphone pedestrian holds. However, the assumption may be wrong in many cases because people usually hold on their smartphone tilted. The gap between the heading of the smartphone and the actual heading of the pedestrian becomes larger as the pedestrian walks. This kind of error can be estimated and calibrated utilizing the UWB positioning result. If we collect the results of UWB positioning while the pedestrian walks straight for enough time, we can estimate the actual heading of the pedestrian by using collected data. When the pedestrian keeps on walking straight, the heading of smartphone can be calibrated by using the estimated actual heading of the pedestrian instead of the heading of smartphone calculated by PDR. Algorithm 1 summarizes the proposed heading estimation.

As seen in Algorithm 1, we define $\mathbf{P}_{\text{str}}$ as the set of all UWB position when the user walks straight, and EV as the estimated heading vector. After the initialization, inertial heading at time t, ih_t is calculated in the same way that the PDR calculates heading using gyroscope and magnetometer data. When the difference between previous inertial heading and current inertial heading is smaller than the pre-defined threshold ih_τ, the pedestrian is considered to be walking straight (line 5), and in this situation, we collect result of UWB positioning P_t. The result of UWB positioning P_t is stored in $\mathbf{P}_{\text{str}}$, which is the set of all results of UWB positioning when the pedestrian walk straight. Simultaneously, the estimated heading vector EV is calculated by adding all heading vectors of $\mathbf{P}_{\text{str}}$ (line 7-9). To reduce the error caused by the outlier of positioning results, we assume that the sum of all the heading vectors corresponding to the positioning is the heading vector collected when the pedestrian walks straight. If we collect enough UWB positioning-based positions to detect heading,

Authorized licensed use limited to: ULAKBIM UASL - Galatasaray Universitesi. Downloaded on December 23,2024 at 21:46:19 UTC from IEEE Xplore. Restrictions apply.

Algorithm 1 UWB positioning-assisted heading estimation

1: $\mathbf{P}_{\text{str}} = \emptyset$
2: $EV = (0, 0)$
3: **for** t=1,2,... **do**
4: Let P_t be the UWB positioning result at time t
5: **if** $ih_t - ih_{t-1} \leq ih_\tau$ **then**
6: $\mathbf{P}_{str} = \mathbf{P}_{\text{str}} \bigcup P_t$
7: **for** all P in $\mathbf{P}_{\text{str}}$ **do**
8: $EV+ = P_t - P$
9: **end for**
10: **if** $|\mathbf{P}_{\text{str}}| \geq l_\tau$ **then**
11: $Heading = \arctan(EV.y/EV.x)$
12: **else**
13: $Heading = ih_t$
14: **end if**
15: **else**
16: $\mathbf{P}_{\text{str}} = \emptyset$
17: $EV = (0, 0)$
18: **end if**
19: **end for**

we can calculate $Heading$ by taking $\arctan(EV.y/EV.x)$, which $EV.x$, $EV.y$ are the x, y coordinates of the estimated heading vector, respectively. Otherwise, $Heading$ is considered as ih_t (line 10-14). $\mathbf{P}_{\text{str}}$ and EV are initialized when the pedestrian does not seem to be walking straight because the device heading is changed (line 15-18).

B. KF-based UWB/U-PDR combining algorithm

Although the proposed U-PDR algorithm is more accurate than legacy PDR algorithms, PDR algorithms inherently have vulnerability to accumulative error. For this reason, positioning results from the PDR will be incorrect as times go by. On the other hand, UWB positioning is accurate substantially, but in NLOS environment, positioning error can be larger because the UWB ranging values may be longer than the actual distance between the target user and a UWB anchor. These different kinds of errors can be mitigated by compensating each other. The accumulative error of U-PDR can be corrected using the absolute positioning of UWB, and UWB positioning error in NLOS environment can be corrected by the relative position of the proposed U-PDR. The results from each algorithm are combined by KF algorithm to overcome the shortcomings.

In KF, results of the U-PDR positioning are used as estimates, whereas results of the UWB positioning are used as measurements. The position and the velocity calculated by U-PDR are described as the state variables X_t of the filter algorithm, namely,

$$X_t = [x(t), v_x(t), y(t), v_y(t)] \tag{2}$$

where $x(t)$ and $y(t)$ indicate the spatial coordinates while $v_x(t)$ and $v_y(t)$ mean the velocities with respect to the x-axis and y-axis, respectively. Then, the state transition equation and the state transition matrix F for the proposed KF algorithm is defined as equation as follows:

$$\begin{cases} x(t+1) = x(t) + v_x(t)\Delta t \\ y(t+1) = y(t) + v_y(t)\Delta t, \end{cases} \tag{3}$$

$$F = \begin{bmatrix} 1 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \Delta t \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4}$$

It is assumed that the initial position is to be known, and the initial velocity is calculated using the proposed U-PDR. In the prediction step, we predict the position at the next time using the state transition equations mentioned above and result of UWB positioning is measured after prediction. Then, the measurement is calibrated by U-PDR prior to the update step as raw result of the UWB positioning may be incorrect in the case of NLOS environments. The calibrated measurement is used as the measurement in the update step. The KF algorithm updates the state with the measurement. The velocities that will be used in the prediction step are also updated as the following equations.

$$\begin{cases} v_x(t) = S_t \cos\theta_t \\ v_y(t) = S_t \sin\theta_t \end{cases} \tag{5}$$

Finally, the KF algorithm estimates the position of a user by combining the measurement value and the estimated value. Algorithm 2 summarizes the proposed UWB/U-PDR combined KF positioning method. The detailed description of Algorithm 2 is as follows.

Algorithm 2 KF-based UWB/U-PDR combining algorithm

State prediction
$isCorrect = true$
for i=1,2,... **do**
$ur_i =$ the UWB ranging distance of A_i
$pr_i = Dist(P_t, A_i)$
if $ur_i \leq pr_i$ **then**
$isCorrect = false$
end if
end for
if $isCorrect == false$ **then**
Set cr to ur
else
for i=1,2,... **do**
if $ur_i - pr_i \geq r_\tau$ **then**
$cr_i = pr_i$
else
$cr_i = ur_i$
end if
end for
end if
Set update position $P_t = trilaterate(cr)$
State update with $P_t, v_x(t), v_y(t)$

Authorized licensed use limited to: ULAKBIM UASL - Galatasaray Universitesi. Downloaded on December 23,2024 at 21:46:19 UTC from IEEE Xplore. Restrictions apply.

In Algorithm 2, we define P_t, A_i, ur, and cr where P_t is the predicted position at time t by KF, A_i is the ith UWB anchor, ur is the UWB ranging value of all UWB anchors, and cr is the calibrated ranging value of all UWB anchors. After we set $isCorrect$ flag to true, we get ur_i and pr_i, which are UWB ranging value and the Euclidean distance between P_t and A_i ($Dist(P_t, A_i)$), for each ith UWB anchor A_i (line 2-5). Then, we compare ur_i and pr_i to check P_t is predicted properly. In the real world, UWB ranging value is always longer than the actual range value. That is, if the distance between the predicted position and UWB anchor is longer than UWB ranging value, it means the prediction is wrong. So if ur_i is smaller than pr_i, set the flag to false because the predicted position P_t is not expected to be the right value. We check all pr_i for possible anchors in order to check the validity of prediction (line 6-9). After the loop, $isCorrect$ flag is checked. If the flag is set as false, it means that P_t is inaccurate. Thus, ur_i cannot be calibrated based on the P_t, so we set calibrated range cr_i to be same with ur_i (line 10-11). If not, P_t is thought to be accurate. Then we compare ur_i and pr_i again. If ur_i is too longer than pr_i, which means UWB ranging is not accurate because of NLOS situation, so cr_i is set to the ranging value corresponding to the predicted position pr_i. Otherwise, cr_i is set to ur_i (line 12-20). Finally, we can get the position by multilateration of cr_is, and use it as a position input for the update step, P_t. The state is updated with P_t and velocities $v_x(t), v_y(t)$ (line 21-22).

III. Experimental results

Experiments are conducted to evaluate the performance of each proposed algorithm precisely. To evaluate the performance of proposed deep-learning based speed estimation, it is compared with traditional approaches of PDR proposed in [13] and [14]. For the proposed heading estimation, we measure the performance of our algorithm comparing with that of the results from not calibrated heading calculation. Finally, KF fusion algorithm is compared with UWB positioning algorithm and U-PDR algorithm, to show our fusion algorithm performs better than each algorithm only.

A. Experimental settings

In experiments, the subject walks holding a smartphone attached with a UWB tag module at chest height. The experiments are conducted in the hallway located on the third floor of the college of engineering building 302 in Seoul National University campus. The eight anchor nodes are deployed at the experiment place in advance. Samsung Galaxy S7 (SM-G930S) is selected as the experimental mobile smartphone and Decawave DWM1001 is used as the UWB tag module. All 9-axis inertial sensors which equipped in the smartphone are used in our experiment, and DWM1001 uses UWB channel 5, which is a 6.5 GHz band with a bandwidth of 500 MHz. The measured error of UWB ranging of DWM1001 module is less than 10 cm in line of sight condition [15].

B. Deep learning-based speed estimation

For this experiment, training data is collected while the subject is wandering the hallway in the building for an hour. $36,000$ samples are used for training in the offline phase. On the online phase, the subject walks straight at 58.5 m long hallway. The difference between the real distance (i.e., 58.5 m) and estimated distance the subject travels by the model was measured. The experiment is performed for three different speeds (slow, normal, and fast), and ten times repeated per each speed. The results are shown in Table I.

TABLE I
THE AVERAGE DIFFERENCE OF DISTANCES W.R.T. VARIABLE SPEED

	Slow	Normal	Fast	Total
Proposed	2.93 m	4.32 m	4.11 m	3.33 m
Weinberg [13]	10.22 m	9.12 m	9.40 m	9.95 m
Wonho Kang [14]	3.03 m	2.81 m	18.28 m	8.11 m

The model used for comparison is a traditional function model, which regards an input as a difference between maximum acceleration and minimum acceleration while pedestrian is walking to identify the stride lengths of the user. The equation of Weinberg model in [13] is presented as $L_k = \alpha \times \sqrt[4]{A_{d,k}}$ and the equation of Wonho Kang model in [14] is presented as follows,

$$L_k = \begin{cases} \alpha \sqrt[4]{A_{d,k}} + \gamma, & \text{for } A_{d,k} < A_\tau \\ \beta \log(A_{d,k}) + \omega, & \text{for } A_{d,k} \geq A_\tau, \end{cases} \quad (6)$$

where L_k is a stride length at step k, $A_{d,k}$ is difference between maximum acceleration and minimum acceleration in step k, and A_τ is a threshold. α, β, γ, and ω are hyper-parameters needs to be tuned.

As a result, the Wonho Kang model [14] has a similar performance with the proposed scheme at slow and normal speed, whereas it shows lower performance compared to the proposed scheme at fast speed. The Weinberg model [13] always has lower performance compared to the proposed scheme at all different speeds. The experiment shows that the deep learning-based speed estimation outperforms the others.

C. UWB-assisted heading estimation

The subject walks along the edges of a rectangle (5 m $\times$ 31 m) in this experiment. The experiment is repeated four times. The average error of estimated heading with calibration is $8.5°$, which is very smaller than the average error of reference heading without calibration ($16.2°$).

In Fig. 4, the left figure shows PDR trajectory without using the heading calibration and the right one shows PDR trajectory with the proposed heading estimation algorithm. We can see that when the subject goes straight for a certain amount of time, calibration is applied and the positioning result gets more accurate in the proposed algorithm.

D. KF-based UWB/U-PDR combining algorithm

Finally, the performance of the KF-based UWB/U-PDR combining algorithm is evaluated. In this experiment, the

Authorized licensed use limited to: ULAKBIM UASL - Galatasaray Universitesi. Downloaded on December 23,2024 at 21:46:19 UTC from IEEE Xplore. Restrictions apply.

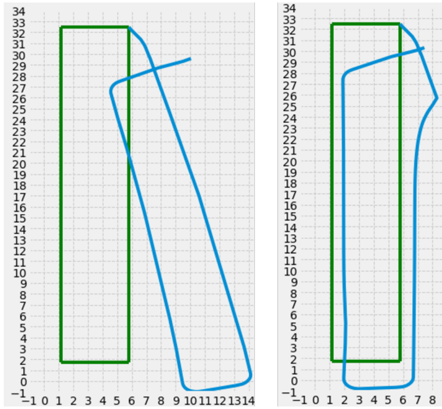

Fig. 4. PDR trajectory with and without heading calibration

subject walks along the same route experimented for the heading estimation (i.e., edges of a rectangle (5 m $\times$ 31 m)), and the experiment is repeated four times.

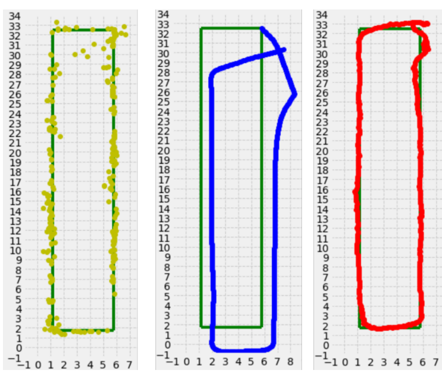

Fig. 5. Positioning results from UWB positioning, U-PDR, and fusion algorithm

In Fig. 5, the leftmost figure shows the result of simple UWB positioning, which is the trilateration result of UWB positioning. In this figure, we can see a few points which are far from the reference line (i.e., green line that the subject walked). This is the influence of the ranging values measured in NLOS environment which reduces the accuracy. Also, positioning is not continuous because of the relatively low rate of UWB ranging. For this reason, simple UWB positioning is not suitable for real-time applications that require high positioning rates. The average error of simple UWB positioning is 0.29 m. The figure in middle illustrates the positioning result of U-PDR. Although the accuracy is better than traditional PDR, it is getting inaccurate as time goes by because of accumulative error. The average error of U-PDR is 1.15 m. The result of the proposed KF fusion algorithm is depicted in the rightmost figure. Under the influence of the fusion of UWB positioning and U-PDR, we can observe that our KF-based fusion algorithm can trace the real path of the user in real-time. Moreover, it is almost free from the NLOS error of UWB positioning and accumulative error of PDR. The average error of the fusion algorithm is 0.21 m. It also shows the fusion algorithm outperforms the others.

IV. Conclusion

In this work, a noble UWB positioning and PDR combined indoor positioning algorithm is proposed. The proposed method contains U-PDR and KF-based UWB positioning and U-PDR fusion algorithm. U-PDR provides the personalized model for PDR by training the distinctive patterns of pedestrians with deep learning. In addition, tilt of smartphone is calibrated when pedestrians walk straight, which improves the accuracy of heading estimation. Experimental results show that the proposed U-PDR performs better than the traditional PDR on average. This is because KF-based UWB positioning and U-PDR combining algorithm decreases positioning error by taking advantages of UWB positioning and U-PDR. Therefore, the fusion algorithm shows its effectiveness in comparison with UWB positioning or PDR only. The proposed positioning scheme can be easily implemented on the smartphone in a real-time scenario. Consequently, because of its high accuracy and availability, the proposed algorithm is expected to be used as an indoor positioning method for various LBS.

References

[1] D. Ahmetovic, C. Gleason, K. M. Kitani, H. Takagi, and C. Asakawa, "NavCog: turn-by-turn smartphone navigation assistant for people with visual impairments or blindness," in *Proceedings of the 13th Web for All Conference*, 2016, pp. 9:1–9:2.

[2] S. Alletto, R. Cucchiara, G. D. Fiore, L. Mainetti, V. Mighali, L. Patrono, and G. Serra, "An indoor location-aware system for an iot-based smart museum," *IEEE Internet of Things Journal*, vol. 3, no. 2, pp. 244–253, Apr. 2016.

[3] N. E. Klepeis et al., "The national human activity pattern survey (NHAPS): a resource for assessing exposure to environmental pollutants," *Journal of Exposure Scient & Environmental Epidemiology*, vol. 11, no. 3, pp. 231–252, July 2001.

[4] L. A. Latiff, A. Ali, O. Chia-Ching, and N. Fisal, "Development of an indoor gps-free self-positioning system for mobile ad hoc network (manet)," in *IEEE Networks Jointly held with IEEE ICC*, Nov. 2005.

[5] Y. Wu, H.-B. Zhu, Q.-X. Du, and S.-M. Tang, "A survey of the research status of pedestrian dead reckoning systems based on inertial sensors," *International Journal of Automation and Computing*, vol. 16, Sept. 2018.

[6] L. L. Shen and W. W. S. Hui, "Improved pedestrian dead-reckoning-based indoor positioning by RSSI-based heading correction," *IEEE Sensors Journal*, vol. 16, no. 21, pp. 7762–7773, Nov. 2016.

[7] C. Fischer et al., "Ultrasound-aided pedestrian dead reckoning for indoor navigation," in *Proc. ACM Int. Workshop Mobile Entity Localization Tracking GPS-Less Environ.*, Sept. 2008, pp. 31–36.

[8] M. Strutu, D. Caspari, J. Pickert, U. Grossmann, and D. Popescu, "Pedestrian smartphone based localization for large indoor areas," in *Proc. IEEE IDAACS*, Sept. 2013, pp. 450–454.

[9] A. R. J. Ruiz, F. S. Granja, J. C. P. Honorato, and J. I. G. Rosas, "Accurate pedestrian indoor navigation by tightly coupling foot-mounted IMU and RFID measurements," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 1, pp. 178–189, Jan. 2012.

[10] P. K. Yoon, S. Zihajehzadeh, B.-S. Kang, and E. J. Park, "Adaptive Kalman filter for indoor localization using Bluetooth low energy and inertial measurement unit," in *IEEE EMBC*, Aug. 2015, pp. 825–828.

[11] P. Chen, Y. Kuang, and X. Chen, "A uwb/improved pdr integration algorithm applied to dynamic indoor positioning for pedestrians," *Sensors*, vol. 17, Sept. 2017.

[12] J. Kang, J. Lee, and D.-S. Eom, "Smartphone-based traveled distance estimation using individual walking patterns for indoor localization," *Sensors*, vol. 18, Sept. 2018.

[13] H. Weinberg, "Using the adxl202 in pedometer and personal navigation applications," *Analog Devices AN-602 application note*, 2.2, 2002.

[14] W. Kang and Y. Han, "Smartpdr: Smartphone-based pedestrian dead reckoning for indoor localization," *IEEE Sensors Journal*, vol. 15, pp. 2906–2916, Dec. 2014.

[15] *DWM1001 System Overview*, Decawave, 2017, version 1.0.

Authorized licensed use limited to: ULAKBIM UASL - Galatasaray Universitesi. Downloaded on December 23,2024 at 21:46:19 UTC from IEEE Xplore. Restrictions apply.