Detection of Pedestrian Turning Motions to Enhance Indoor Map Matching Performance

Seunghyeon Park
School of Integrated Technology
Yonsei University
Incheon, Korea
seunghyeon.park@yonsei.ac.kr

Taewon Kang
School of Integrated Technology
Yonsei University
Incheon, Korea
taewon.kang@yonsei.ac.kr

Seungjae Lee
School of Integrated Technology
Yonsei University
Incheon, Korea
seungjae@yonsei.ac.kr

Joon Hyo Rhee*
Korea Research Institute of Standards and Science
Daejeon, Korea
jh.rhee@kriss.re.kr
* Corresponding author

Abstract—A pedestrian navigation system (PNS) in indoor environments, where global navigation satellite system (GNSS) signal access is difficult, is necessary, particularly for search and rescue (SAR) operations in large buildings. This paper focuses on studying pedestrian walking behaviors to enhance the performance of indoor pedestrian dead reckoning (PDR) and map matching techniques. Specifically, our research aims to detect pedestrian turning motions using smartphone inertial measurement unit (IMU) information in a given PDR trajectory. To improve existing methods, including the threshold-based turn detection method, hidden Markov model (HMM)-based turn detection method, and pruned exact linear time (PELT) algorithm-based turn detection method, we propose enhanced algorithms that better detect pedestrian turning motions. During field tests, using the threshold-based method, we observed a missed detection rate of 20.35% and a false alarm rate of 7.65%. The PELT-based method achieved a significant improvement with a missed detection rate of 8.93% and a false alarm rate of 6.97%. However, the best results were obtained using the HMM-based method, which demonstrated a missed detection rate of 5.14% and a false alarm rate of 2.00%. In summary, our research contributes to the development of a more accurate and reliable pedestrian navigation system by leveraging smartphone IMU data and advanced algorithms for turn detection in indoor environments.

Index Terms—Pedestrian navigation system, pedestrian turn detection, hidden Markov model, pruned exact linear time, indoor map matching.

I. Introduction

A pedestrian navigation system (PNS) in indoor environments [1], [2], where global navigation satellite system (GNSS) [3]–[10] signal access is difficult, is necessary, particularly for search and rescue (SAR) [11]–[13] operations in large buildings. Some systems rely on preinstalled infrastructure. These include pseudolites [14], radio-frequency identifications (RFIDs) [15], [16] and ultra-wideband (UWB) radars [17]. However, these infrastructure installation and maintenance procedures require significant time and effort [18], [19]. To navigate without the use of infrastructure, pedestrian dead reckoning (PDR) systems sometimes use an inertial measurement unit (IMU) mounted on the foot because the foot-mounted IMU has an explicit stance stage that can be used to suppress the long-term drift of inertial sensors using the zero-velocity update (ZUPT) extended Kalman filter (EKF) method [20].

Our work aims to investigate and leverage pedestrian walking behaviors to enhance the performance of indoor PDR and map matching techniques. PDR enables the tracking of a pedestrian's trajectory without relying on external signals or infrastructure. However, PDR is susceptible to cumulative errors over time, leading to a significant degradation in position estimation accuracy. By reliably detecting pedestrian turning motions within a given indoor PDR trajectory, we can identify the locations where turns occur, which can be considered as intersections of corridors. These intersections serve as reference points to improve the accuracy of indoor map matching techniques. Thus, our study focuses on the detection of pedestrian turning motions in the PDR trajectory.

Several methods to detect pedestrian turns have been developed by various researchers. Three representative methods to detect a pedestrian's turning motion are as follows.

1) Threshold-based turn detection method [21]
2) Hidden Markov model (HMM)-based turn detection method [22]
3) Pruned exact linear time (PELT) algorithm-based turn detection method [23]

However, these methods have limited accuracy. There are turns that cannot be detected by these algorithms, and there are cases in which a turn is detected even when it is not a turn. Therefore, we propose methods to improve each of the three turn detection methods to enhance the performance.

II. Threshold-Based Turn Detection Method

A. Existing Threshold-Based Turn Detection Method

The existing threshold-based method detects the turn if the changed value of the yaw angle between adjacent steps
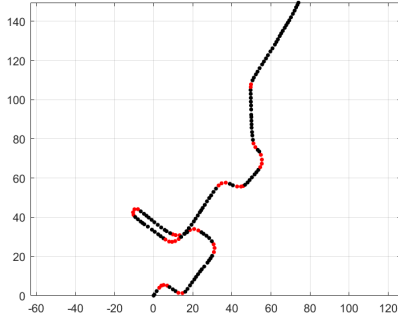
Fig. 1. Map showing pedestrian steps measured by a smartphone IMU and processed by a conventional PDR algortihm.

Fig. 2. Pedestrian walking directions and regions.

Fig. 3. HMM model used for turn detection.

exceeds a predefined threshold. This method was developed in several studies [21], [24], [25] owing to its simplicity and cost-effectiveness. In the example of Fig. 1, the pedestrian walked through an arbitrary indoor route while holding a smartphone, recording motion data of the pedestrian in a time series format using built-in IMU sensors. Then, the data was processed by a conventional PDR algorithm to plot the trajectory of the pedestrian. The dot (black and red) forming the trajectory in Fig. 1 shows the position estimate of each step. The black dots show the steps where the pedestrian walked straight, and the red dots show the steps where the pedestrian turned.

However, this method shows a low detection accuracy when the length of a turn is longer than usual turns. If the turn consists of many steps, the yaw angle changes between adjacent steps in the same turn can vary. Thus, some steps with yaw angle changes exceeding the threshold are declared as turns and the others are not, even though those steps belong to the same turn. As a result, a single turn can be recognized as multiple erroneous turns if this turn-detection method is applied.

B. Proposed Threshold-Based Turn Detection Method

To address the aforementioned weakness of the threshold-based turn detection method, we propose an additional algorithm. When a detected turn step is adjacent to previously detected turn steps, it is reasonable to consider it as part of the existing turn sequence. Therefore, in our algorithm, when multiple turns are in close proximity to each other, they are treated as a single turn. This approach effectively reduces the number of erroneously detected turns. The performances of the existing and proposed methods are compared in Section V-B.

III. HIDDEN MARKOV MODEL (HMM)-BASED TURN DETECTION METHOD

A. Existing HMM-Based Turn Detection Method

The hidden Markov model (HMM) is used to deduce input information by statistically analyzing large amounts of output data [26], [27]. In order for HMM to be employed in modeling the pedestrian's walking behavior, the pedestrian's prior step information needs to be maintained to some extent. Therefore, Gu *et al.* [23] utili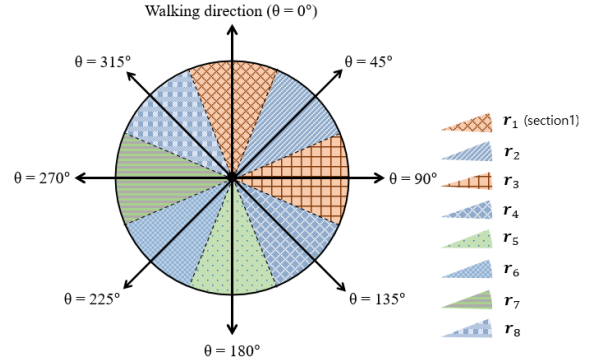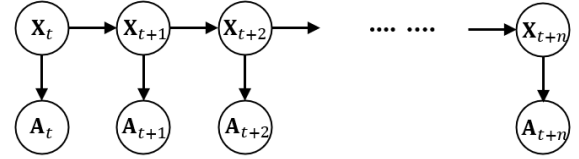zed the HMM algorithm by combining three walking steps as a single state of HMM assuming that a single turning motion consists of three walking steps. In this manner, this method can analyze the relationship between the six adjacent steps (i.e., two states of HMM), rather than two adjacent steps, while using the HMM algorithm for detecting a turning motion.

Specifically, the existing method [23] classifies the walking direction into eight regions according to the angle of the direction, as shown in Fig. 2. The input used in this classification process is the walking direction angle θ for each step, and then the steps are classified into region r to which θ belongs. Since there are many possible values of a state with three walking steps (i.e., $8 \times 8 \times 8 = 256$ possible values), values that do not appear and values that appear at low frequency are removed, and only 14 values (i.e., $v_1, \ldots, v_{14}$) are configured through clustering in this approach.

This method [23] does not detect or calculate the probability of the pedestrian turning on the following step; instead, it predicts the probability of being in each of the 14 values (i.e., $v_1, \ldots, v_{14}$). In Fig. 3, the observation matrix A_t contains $P(v_1), P(v_2), \ldots, P(v_{14})$, which represents the probabilities of each value that the state X_{t+1} can have after the state X_t.

Therefore, to perform turn detection based on the information obtained from the existing HMM technique [23], we utilized the probabilities of the possible values for the next state X_{t+1}, given the value of the current state X_t. For instance, suppose X_t has the value v_1 and $P(v_1) = 0.6, P(v_2) = 0.01, \ldots, P(v_{14}) = 0.1$. In this scenario, X_{t+1} has a 60% probability of having the same value as X_t due to $P(v_1) = 0.6$. If the probability $P(v_t)$ of maintaining the current value v_t of state X_t is lower than 50%, it is determined as a turn since the state value is likely to change with a probability greater than 50%. Conversely, if

the probability of maintaining the current state value is higher than 50%, it is determined that the pedestrian does not make a turn for that state.

However, as depicted in Fig. 1, the process of turning or walking straight does not always occur in three steps. There are instances where a turn is completed in one or two steps, and in some cases, a turn may take more than six steps. Consequently, the existing HMM approach exhibits limited accuracy in turn detection. To achieve more effective modeling, it becomes necessary to capture the broader human gait behavior. In this regard, we propose a modified HMM algorithm, which will be elaborated upon in the subsequent subsection.

B. Proposed HMM-Based Turn Detection Method

In the proposed HMM-based turn detection method, we group steps exhibiting similar behavior into a new entity. This grouping is necessary because the number of steps required for a single turn is not always fixed, as assumed in [23], where three steps were considered. In this paper, we refer to this linked group of steps as a "block." Consequently, a block can consist of as few as one step or as many as ten steps, for example. In our HMM, each individual state corresponds to a single block. To determine the length of each block, we employed the Butterworth bandpass filter. This filter identifies the peak locations of yaw angle measurements, which can be considered as change points in a pedestrian's walking behavior. The time interval between these peaks is then selected as the length of a block. This approach enhances the accuracy of comprehending pedestrian turning behavior in comparison to existing methods.

The main feature of a pedestrian's trajectory is the direction angle θ. This angle, θ, is referred to as the difference between the body's yaw plane velocity vector and orientation vector. We computed the orientation vector using the smartphone IMU's gyroscope measurements and obtained the velocity vector using its acceleration measurements. We categorized θ into eight discrete state values based on the corresponding regions for each step, as shown in Fig. 2. Therefore, a pedestrian's movement is considered a turn only when it deviates more than $22.5°$ from the current direction.

The transition matrix and emission matrix, which are modeling parameters of the HMM, are constructed using the eight possible state values. The process of constructing the parameters of HMM is as follows:

First, we obtain the region number for each block. There are eight regions denoted as $\{r_1, r_2, r_3, \ldots, r_8\}$ as shown in Fig. 2. This region number is directly determined based on the θ information obtained earlier. For example, if $22.5° \leq \theta < 67.5°$, the region number for that state or block is r_2. The region of the pedestrian's first block is designated as r_1 and serves as the baseline for the entire walk. If the pedestrian turns 30 degrees clockwise from the initial walking direction, for example, the region number changes from r_1 to r_2.

Secondly, we calculate the transition probabilities. In the Markov chain, there are several states, and the transition probability refers to the probability of moving from one state to another [28], [29]. Therefore, we calculated the probability of transitioning from one of the eight regions $\{r_1, r_2, r_3, \ldots, r_8\}$ to another, as presented in Table I.

The third step is to obtain the emission probabilities, which represent the probability distribution of observing evidence A_t in a given state X_t. Each state X_t corresponds to one of the eight regions shown in Fig. 2. The evidence A_t refers to the status of the pedestrian, indicating whether they are turning or walking straight. Consequently, the emission matrix represents the probabilities of the pedestrian turning or not turning at each region, as listed in Table II.

A turn is detected if the probability of staying in the current state value at the following state is less than 0.5. The performance of both the existing and proposed methods is compared in Section V-B.

IV. PRUNED EXACT LINEAR TIME (PELT) ALGORITHM-BASED TURN DETECTION METHOD

A. Existing PELT-Based Turn Detection Method

To detect a turn, it is necessary to define the point at which the turn occurs, indicating a rapid change in the statistical properties [22], [30]. In the pruned exact linear time (PELT) approach, pruning is carried out at the current time step if the current cost is less than the cost at the potential change-point plus the extra segment cost. For a given set of data $X = (x_1, x_2, ..., x_n)$, let $t_0 = 0$, $t_{k+1} = n$, and $t_1, t_2, ..., t_k$ be the points of change in chronological order. These points of change are the ones that minimize the following equation [22]:

$$\sum_{i=1}^{k+1} \left(C\left(x_{t_{i-1}+1:t_i}\right) + \beta \right) \tag{1}$$

where $x_{a:b} = (x_a, x_{a+1}, ..., x_b)$ $(a \leq b)$, $C(x_{a:b})$ is the cost function of $x_{a:b}$, and β represents the penalization to prevent overfitting.

However, in the PELT approach, there were instances where a genuine turn was missed, occurring in 44.67% of the cases. To enhance the performance of the model, an anomaly detection algorithm was incorporated to improve its capabilities. Anomaly detection involves identifying rare observations that significantly deviate from the majority of the data and do not conform to normal behavior. By incorporating this algorithm, the model becomes more adept at detecting such anomalies and improving overall performance.

B. Existing IF-Based Turn Detection Method

In our study, the isolation forest (IF) algorithm was utilized for anomaly detection. The isolation forest operates on a tree-based structure by randomly partitioning the data to isolate all observations, as shown in Fig.4 [31]–[33]. The algorithm is based on the simple principle that isolating anomalies is easier than isolating normal data [34], [35]. Outliers tend to be closer to the root node of the tree, resulting in shorter path lengths [36], [37]. By employing a collection of trees that are randomly fitted to the data, we can calculate the average depth

region	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8
r_1	0.5921	0.0542	0.0244	0.0105	0.0294	0.0114	0.0162	0.0705
r_2	0.0950	0.7808	0.1389	0.0056	0.0037	0.0030	0.0015	0.0041
r_3	0.0257	0.1220	0.7311	0.0670	0.0074	0.0000	0.0015	0.0062
r_4	0.0515	0.0079	0.0944	0.8406	0.1532	0.0068	0.0015	0.0041
r_5	0.0277	0.0068	0.0044	0.0664	0.6949	0.0781	0.0103	0.0073
r_6	0.0416	0.0090	0.0011	0.0050	0.1029	0.8225	0.1521	0.0093
r_7	0.0238	0.0056	0.0000	0.0019	0.0049	0.0698	0.7061	0.0870
r_8	0.1426	0.0102	0.0056	0.0031	0.0037	0.0083	0.1108	0.8104

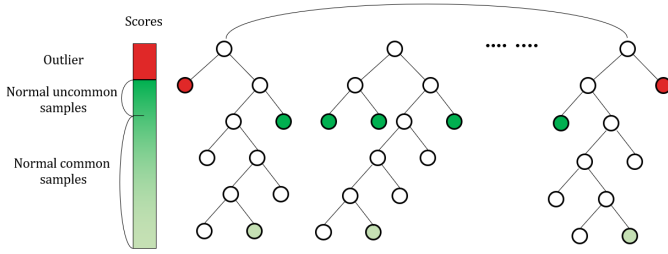| turn	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8
O	0.0812	0.1085	0.1300	0.1011	0.1360	0.0979	0.1773	0.1192
X	0.9188	0.8915	0.8700	0.8989	0.8640	0.9021	0.8227	0.8808

Fig. 4. Tree of isolation forest.

at which outliers occur within the trees. This average depth is then used to generate a final score reflecting the degree of "outlierness" for each data point.

By utilizing the IF algorithm, outliers can be effectively detected. We considered nine types of IMU data, including acceleration, gravitational force, and magnetic field along the x, y, and z axes, as the input data. It was observed that there are differences in this data within steps that involve turning [38]. Therefore, turns can be detected through the IF algorithm, which detects outliers from the given data.

C. Proposed PELT+IF-Based Turn Detection Method

We employed the IF algorithm to address the limitations of the PELT approach. When using the PELT algorithm, the false alarm rate was low (6.97%), but there was a significant missed detection rate (44.67%). In cases where an outlier exhibited distinct characteristics, it was challenging for the PELT algorithm to recognize it as a "point of change." However, it is highly probable that such outliers represent actual points of change, thus necessitating their detection through the use of the IF algorithm.

In the IF algorithm, a scoring process is employed to determine the extent to which a separated point should be considered an outlier. This scoring process helps establish a threshold for identifying outliers based on their deviation from the original data. The scoring calculation in the IF algorithm is performed using the following equation [32]:

$$s\left(x, n\right) = 2^{-\frac{E(h(x))}{c(n)}} \tag{2}$$

Here, $h(x)$ represents the path length of the corresponding observation, $E(h(x))$ denotes the average path length for that observation across all iterations, and $c(n)$ represents the average path length of the tree used to determine $h(x)$.

For normal data, where the observed value x is similar to the average path length, $E(h(x))$ converges to $c(n)$, resulting in a score of approximately 0.5 [39], [40]. However, if the observed value x is an outlier, $E(h(x))$ tends to converge to 0, leading to a score close to 1. Since the maximum path length of the observation x is $E(h(x)) = n-1$, the score $s(x, n)$ falls within the range of 0 to 1. A score greater than 0.5 indicates a higher likelihood of being an outlier, while a score less than or equal to 0.5 indicates normal data. The pollution parameter in IF [41] controls the threshold for determining when a scored data point should be considered an outlier. We experimentally set the pollution parameter to 0.05 to achieve the best result.

V. TEST RESULTS

A. Data Collection

A total of 58 instances of sensor log data collected in the laboratory were utilized as both input and output data values. These nine types of IMU data encompassed the acceleration along the x-axis, acceleration along the y-axis, acceleration along the z-axis, gravitational force along the x-axis, gravitational force along the y-axis, gravitational force along the z-axis, magnetic field along the x-axis, magnetic field along the y-axis, and magnetic field along the z-axis.

B. Experiment Results

The performance of the proposed methods in Sections II-B, III-B, and IV-C was evaluated by analyzing missed detections and false alarms. Missed detection refers to cases where a turn

TABLE III
PERFORMANCE COMPARISON BETWEEN TWO THRESHOLD-BASED ALGORITHMS IN TURN DETECTION

	Missed detection	False alarm
Existing algorithm	20.35%	10.45%
Proposed algorithm	20.35%	7.65%

TABLE IV
PERFORMANCE COMPARISON BETWEEN TWO HMM-BASED ALGORITHMS IN TURN DETECTION

	Missed detection	False alarm
Existing algorithm	7.11%	30.33%
Proposed algorithm	5.14%	2.00%

TABLE V
PERFORMANCE COMPARISON BETWEEN TWO PELT-BASED ALGORITHMS IN TURN DETECTION

	Missed detection	False alarm
Existing algorithm	44.67%	6.97%
Proposed algorithm	8.93%	6.97%

is not recognized among all the actual turns. On the other hand, false alarms occur when a turn is incorrectly identified even though no rotation has taken place. The occurrence of these cases was quantified as probabilities.

1) Threshold-Based Turn Detection Method Performance: Among the 159 collected ambulation data, we computed the missed detection and false alarm rates for accurately detecting 1,979 executed turns. In the existing threshold-based turn detection method, the missed detection rate was determined to be 20.35%, while the false alarm rate was 10.45%, as indicated in Table III. For the proposed threshold-based turn detection method, the missed detection rate was found to be 20.35%, and the false alarm rate was 7.65%.

2) HMM-Based Turn Detection Method Performance: The performance of the HMM methods is presented in Table IV. When using the existing HMM algorithm, the missed detection rate was 7.11%, and the false alarm rate was 30.33%. This algorithm is considered unreliable as it frequently identifies steps as turns, even when no actual turns are performed. In contrast, the proposed HMM algorithm achieved a missed detection rate of 5.14%, and a false alarm rate of 2.00%, which is significantly lower than the existing method. Particularly, there was a significant improvement in the false alarm rate by the proposed method.

3) PELT+IF-Based Turn Detection Method Performance: The missed detection rate accounted for 44.67% of the total 1,979 turns, while the false alarm rate was 6.97%. In the case of the PELT-based turn detection method, a noticeable "overlap" occurred, where rotations that had already been detected were detected again. This overlap occurred 305 times out of a total of 1,979 turns. Notably, overlaps tend to occur when there is a prolonged change in the yaw angle, making it inappropriate to consider it as a single point. As there are many instances of overlap where the same point is recognized as multiple turns, the missed detection rate significantly increases.

To evaluate the effectiveness of the added IF algorithm for reducing the missed detection rate, we examined its ability to detect the missed detection portion of the original PELT algorithm. Using the same dataset, 80% of the missed detection cases of PELT were accurately detected as turns. This led to a reduced missed detection rate of 8.93%, while the false alarm rate remained at 6.97%.

VI. CONCLUSION

In this study, we proposed three new approaches to detect pedestrian turning motions based on smartphone IMU information. The limitations of existing methods in turn detection were investigated. Among the three proposed algorithms, the HMM-based algorithm demonstrated the best performance with a missed detection rate of 5.14% and a false alarm rate of 2.00%. This turn detection algorithm can be applied to enhance indoor map matching performance for pedestrian navigation systems.

ACKNOWLEDGMENT

REFERENCES

[1] T. Kang and Y. Shin, "Indoor navigation algorithm based on a smartphone inertial measurement unit and map matching," in *Proc. ICTC*, 2021, pp. 1421–1424.
[2] J. Park, S. Jang, and Y. Shin, "Indoor path planning for an unmanned aerial vehicle via curriculum learning," in *Proc. ICCAS*, 2021, pp. 529–533.

[3] M. Kim, J. Seo, and J. Lee, "A comprehensive method for GNSS data quality determination to improve ionospheric data analysis," *Sensors*, vol. 14, no. 8, pp. 14 971–14 993, Aug. 2014.

[4] H. Lee, S. Pullen, J. Lee, B. Park, M. Yoon, and J. Seo, "Optimal parameter inflation to enhance the availability of single-frequency GBAS for intelligent air transportation," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 17 801–17 808, Oct. 2022.

[5] D. De Lorenzo, S. Lo, J. Seo, Y.-H. Chen, and P. Enge, "The WAAS/L5 signal for robust time transfer: Adaptive beamsteering antennas for satellite time synchronization," in *Proc. ION GNSS*, 2010, pp. 2106–2116.

[6] K. Park and J. Seo, "Single-antenna-based GPS antijamming method exploiting polarization diversity," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 2, pp. 919–934, Apr. 2021.

[7] Y.-H. Chen, J.-C. Juang, D. De Lorenzo, J. Seo, S. Lo, P. Enge, and D. Akos, "Real-time dual-frequency (L1/L5) GPS/WAAS software receiver," in *Proc. ION GNSS*, 2011, pp. 767–774.

[8] Y. Lee, Y. Hwang, J. Y. Ahn, J. Seo, and B. Park, "Seamless accurate positioning in deep urban area based on mode switching between DGNSS and multipath mitigation positioning," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 6, pp. 5856–5870, Jun. 2023.

[9] S. Kim and J. Seo, "Machine-learning-based classification of GPS signal reception conditions using a dual-polarized antenna in urban areas," in *Proc. IEEE/ION PLANS*, Apr. 2023, pp. 113–118.

[10] W. Kim and J. Seo, "Low-cost GNSS simulators with wireless clock synchronization for indoor positioning," *IEEE Access*, vol. 11, pp. 55 861–55 874, 2023.

[11] H. Lee, T. Kang, S. Jeong, and J. Seo, "Evaluation of RF fingerprinting-aided RSS-based target localization for emergency response," in *Proc. IEEE VTC*, Jun. 2022, pp. 1–7.

[12] H. Lee and J. Seo, "Performance evaluation and hybrid application of the greedy and predictive UAV trajectory optimization methods for localizing a target mobile device," in *Proc. ION ITM*, Jan. 2023, pp. 161–171.

[13] ——, "Performance comparison of numerical optimization algorithms for RSS-TOA-based target localization," in *Proc. IEEE VTC*, Jun. 2023, pp. 1–6.

[14] M. Angermann, M. Frassl, M. Doniec, B. J. Julian, and P. Robertson, "Characterization of the indoor magnetic field for applications in localization and mapping. in indoor positioning and indoor navigation," in *Proc. IPIN*, 2012, pp. 1–9.

[15] Fan, Yuanyuan, Liang, and Qingzhong, "An improved method for detection of the pedestrian flow based on RFID," in *Proc. IEEE CSE*, 2017, pp. 69–72.

[16] Q. Liu, P. Liang, Y. Zhou, and X. Lao, "Pedestrian movement monitoring based on face recognition and RFID recognition," in *Proc. ICCSNT*, 2015, pp. 673–676.

[17] Y. Shin, S. Lee, and J. Seo, "Autonomous safe landing-area determination for rotorcraft UAVs using multiple IR-UWB radars," *Aerosp. Sci. Technol.*, vol. 69, pp. 617–624, Oct. 2017.

[18] H. Lee, J. Seo, and Z. Kassas, "Urban road safety prediction: A satellite navigation perspective," *IEEE Intell. Transp. Syst. Mag.*, vol. 14, no. 6, pp. 94–106, Nov.-Dec. 2022.

[19] S. Jeong, H. Lee, T. Kang, and J. Seo, "RSS-based LTE base station localization using single receiver in environment with unknown path-loss exponent," in *Proc. ICTC*, 2020, pp. 958–961.

[20] S. Godha and G. Lachapelle, "Foot mounted inertial system for pedestrian navigation," *Meas. Sci. Technol.*, vol. 19, no. 7, pp. 1–9, 2008.

[21] M. Pham, M. Elshehabi, L. Haertner, S. Del Din, K. Srulijes, T. Heger, M. Synofzik, M. Hobert, G. Faber, C. Hansen, D. Salkovic, J. Ferreira, D. Berg, A. Sanchez-Ferro, J. van Dieën, C. Becker, L. Rochester, G. Schmidt, and W. Maetzler, "Validation of a step detection algorithm during straight walking and turning in patients with Parkinson's disease and older adults using an inertial measurement unit at the lower back," *Front. Neurol.*, vol. 8, pp. 1–9, 2017.

[22] X. Zhou and M. Franklin, "Indoor route and location inference using smartphone IMU sensors," in *Proc. IEEE PIMRC*, 2020, pp. 1–7.

[23] Y. Gu, Q. Song, Y. Li, and M. Ma, "Foot-mounted pedestrian navigation based on particle filter with an adaptive weight updating strategy," *J. Navig.*, vol. 68, no. 1, pp. 23–38, 2015.

[24] A. Gal, T. Douglas, A. Chan, and D. Hay, "Potential for autonomous linear, curvilinear, and phase detection in natural context para skating using IMU sledge motion data," in *Proc. IEEE MeMeA*, 2020.

[25] D. Novak, M. Goršič, J. Podobnik, and M. Munih, "Toward real-time automated detection of turns during gait using wearable inertial measurement units," *Sensors*, vol. 14, no. 10, pp. 18 800–18 822, 2014.

[26] M. Johansson and T. Olofsson, "Bayesian model selection for Markov, hidden Markov, and multinomial models," *IEEE Signal Process. Lett.*, vol. 14, no. 2, pp. 129–132, 2007.

[27] H. I. Turhan, D. Acar, N. B. Ayana, K. Ahıska, and M. Demirekler, "Hidden Markov model based variable structured multiple model algorithm," in *Proc. IEEE SIU*, 2020, pp. 1–4.

[28] G. Ye, D. Chen, and B. Mak, "Transition probabilities are more important than we once thought," in *Proc. IEEE ICASSP*, 2012, pp. 4809–4812.

[29] Guogang, Gaozheng, Wushiqian, and Yumin, "Using hidden Markov model to predict the potential intent of user's gaze behavior," in *Proc. MLISE*, 2021, pp. 38–41.

[30] R. Killick, P. Fearnhead, and I. A. Eckley, "Optimal detection of changepoints with a linear computational cost," *J. Am. Stat. Assoc.*, vol. 107, no. 500, pp. 1590–1598, 2012.

[31] C.-H. Lee, X. Lu, X. Lin, H. Tao, Y. Xue, and C. Wu, "Anomaly detection of storage battery based on isolation forest and hyperparameter tuning," in *Proc. CFMAI*, 2020, pp. 229–233.

[32] K. Zhang, X. Kang, and S. Li, "Isolation forest for anomaly detection in hyperspectral images," in *Proc. IEEE IGARSS*, 2019, pp. 437–440.

[33] Z. Cheng, C. Zou, and J. Dong, "Outlier detection using isolation forest and local outlier factor," in *Proc. ACM RACS*, 2019, pp. 161–168.

[34] C. Wang, "Research on outlier detection of data based on machine learning," in *Proc. ACM TURC*, 2021, pp. 200–203.

[35] L. Zhang and L. Liu, "Data anomaly detection based on isolation forest algorithm," in *Proc. ICCBE*, 2022, pp. 87–89.

[36] Y. Chabchoub, M. U. Togbe, A. Boly, and R. Chiky, "An in-depth study and improvement of isolation forest," *IEEE Access*, vol. 10, pp. 10 219–10 237, 2022.

[37] P. Raumonen, B. Brede, A. Lau, and H. Bartholomeus, "A shortest path based tree isolation method for UAV LiDAR data," in *Proc. IEEE IGARSS*, 2021, pp. 724–727.

[38] P. Zhang, F. He, H. Zhang, J. Hu, X. Huang, J. Wang, X. Yin, H. Zhu, and Y. Li, "Real-time malicious traffic detection with online isolation forest over sd-wan," *IEEE Transactions on Information Forensics and Security*, vol. 18, pp. 2076–2090, 2023.

[39] D. Sun, Y. Hu, Z. Shi, G. Xu, and W. Zhou, "An efficient anomaly detection framework for electromagnetic streaming data," in *Proc. ICBDC*, 2019, pp. 151–155.

[40] D. Xu, Y. Wang, Y. Meng, and Z. Zhang, "An improved data anomaly detection method based on isolation forest," in *Proc. ISCID*, 2017, pp. 287–291.

[41] J. Kim, H. Choi, J. Shin, and J. T. Seo, "Study on anomaly detection technique in an industrial control system based on machine learning," in *Proc. ACM ICEA*, 2021.