Computers, Materials & Continua

Tech Science Press

https://doi.org/10.32604/cmc.2026.082726

ARTICLE

A Deep Learning Approach to Pedestrian Dead Reckoning: Accounting for Latent Variables with Deep Belief Networks

Kyeonghyun Yoo[#], **Sangmin Lee**[#] and **Hwangnam Kim**[*]

Department of Electrical Engineering, Korea University, Seoul, Republic of Korea

*Corresponding Author: Hwangnam Kim. Email: hnkim@korea.ac.kr

[#]These authors contributed equally to this work

Received: 21 March 2026; Accepted: 04 June 2026; Published: 23 July 2026

ABSTRACT: Inertial Measurement Unit (IMU)-based Pedestrian Dead Reckoning (PDR) enables infrastructure-free indoor positioning and requires heading-drift compensation associated with gyroscope bias. This paper proposes a stationary-window-based Deep Belief Network (DBN) framework that learns a gyroscope bias representation from z-axis angular-velocity and sampling-interval sequences observed during stationary intervals and applies it to heading correction during walking. The learned representation captures the residual angular-velocity offset under stationary conditions and serves as an adaptive correction term for subsequent heading integration. Experiments on short-term, three-lap long-term, and complex indoor paths show that stationary-window-based DBN bias estimation is particularly effective in accumulated-drift regimes, such as long-term repeated walking and complex multi-turn trajectories. The proposed DBN method achieved an average absolute trajectory error (ATE) of 3.9587 m in the long-term experiment and 0.9126 m in the complex path experiment. Stationary detection sensitivity analysis further shows that false-positive stationary decisions have a stronger influence on trajectory consistency than false-negative stationary decisions. These results show that the proposed framework maintains stable waypoint-level trajectory behavior in long-term and complex PDR scenarios where heading drift becomes more pronounced.

KEYWORDS: Indoor positioning system; pedestrian dead reckoning; machine learning; deep belief network

1 Introduction

In indoor environments, signal reception is limited, making it generally impossible to perform positioning using the Global Positioning System (GPS). To overcome this limitation, various indoor positioning techniques have been studied, among which Pedestrian Dead Reckoning (PDR) using Inertial Measurement Units (IMUs) has the advantage of enabling positioning without external infrastructure [1–4].

However, PDR has a fundamental limitation in that errors accumulate over time. In particular, gyroscope bias and drift are continuously integrated during heading estimation, which leads to a rapid increase in positioning error as walking time increases [5,6]. This issue becomes more pronounced in long-term walking or in complex path environments involving multiple direction changes.

Existing studies have proposed filtering-based correction, zero-velocity updates, and various sensor fusion methods [5–11]. Recently, learning-based approaches using time-series models such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) have also been studied to model temporal patterns in IMU data [2,3,12]. Although these methods improve trajectory estimation in many cases, long-term heading drift caused by gyroscope bias remains difficult to suppress explicitly.

In this study, we focus on the characteristics of IMUs during stationary states. Ideally, the gyroscope output should be close to zero in stationary conditions, but in practice, a residual angular-velocity offset can remain due to sensor noise, bias, and other latent error factors. Therefore, stationary intervals provide useful observations for characterizing gyroscope bias-related behavior.

To this end, we propose a stationary-window-based Deep Belief Network (DBN) [13] framework that learns a gyroscope bias representation from measurements observed during stationary windows. The learned representation models the residual angular-velocity offset observed during stationary intervals and is used as an adaptive correction term during subsequent walking intervals. This formulation is based on the practical assumption that the stationary-window-derived offset remains approximately valid over short motion intervals.

To evaluate the proposed method, experiments were conducted under short-term, long-term, and complex indoor walking scenarios. The short-term experiment characterizes waypoint-level trajectory behavior under a low-drift reference condition, while the long-term and complex-path experiments evaluate the proposed method under accumulated-drift regimes where heading drift becomes more pronounced. In addition, stationary detection sensitivity analysis characterizes how false-positive and false-negative stationary decisions influence trajectory consistency.

The main contributions of this paper are summarized as follows:

1. We propose a stationary-window-based DBN framework that uses analytically anchored stationary-window supervision to learn a gyroscope bias representation for adaptive heading-drift correction in IMU-based PDR.
2. We evaluate the proposed method under short-term, long-term, and complex indoor walking scenarios and characterize its scenario-dependent trajectory behavior, with stable drift suppression observed in long-term repeated walking and complex multi-turn paths.
3. We analyze stationary detection sensitivity to characterize how false-positive and false-negative stationary decisions influence waypoint-level trajectory consistency.

The remainder of this paper is organized as follows. Section 2 reviews related work on PDR, sensor-fusion-based drift correction, and learning-based inertial positioning. Section 3 presents the proposed system, including the dataset, overall PDR framework, and the proposed stationary-window-based DBN framework. Section 4 describes the experimental settings and evaluates the proposed method through experiments on short-term, long-term, complex-path, and stationary detection error scenarios. Finally, Section 5 concludes the paper and discusses future research directions.

2 Related Work

PDR is a representative IMU-based indoor localization technique that estimates pedestrian position without external infrastructure. A conventional PDR pipeline consists of step detection, step length estimation, and heading integration using accelerometer and gyroscope measurements. Weinberg-type step length models and Kalman-filter-based correction methods are widely used because of their simplicity and low computational cost [1,5,6], but they remain sensitive to accumulated heading error caused by gyroscope noise and bias.

To reduce accumulated error, zero-velocity updates, filtering-based correction, and sensor fusion approaches have been proposed [4,7,11,14]. Fusion-based methods using UWB-IMU, magnetic/GNSS integration, or vision-anchor information can reduce long-term PDR drift [8–10,15–17], but often require additional infrastructure, environmental information, or increased computational resources, limiting their use in lightweight standalone wearable PDR systems.

Learning-based approaches have also been investigated to model nonlinear temporal patterns in IMU data. Sequence models such as LSTM and GRU can learn temporal dependencies from inertial measurements and improve trajectory estimation under certain motion conditions [2,3,12,18,19]. Nevertheless, many learning-based approaches focus on end-to-end trajectory or displacement estimation, and the gyroscope bias component that directly affects heading integration is often treated implicitly rather than explicitly updated.

In contrast, this study focuses on stationary intervals as reliable opportunities for observing gyroscope bias-related behavior. By learning a stationary-window-based bias representation and applying it as a correction term during subsequent walking intervals, the proposed DBN framework directly targets long-term heading drift while maintaining a standalone PDR structure without external infrastructure.

3 Proposed System

3.1 Dataset

The training data were collected using a chest-mounted 6-axis IMU. Data were acquired using a segment-based protocol in which each recording segment was assigned a predefined motion type and duration. This structure was designed to distinguish stationary, transitional, and dynamic motion regimes while preserving the temporal continuity of the overall IMU sequence. For each recording, gyroscope, accelerometer output, and orientation measurements were synchronously stored along with timestamps and segment-level motion labels.

For stationary-bias learning, the collected data were divided into stationary intervals and dynamic intervals using both segment information and motion-state detection results. Consecutive stationary samples were then reconstructed into window-level samples, and a reference bias proxy b_m^* was computed for each stationary window. The DBN input consists of windowed ω_z and dt sequences, while b_m^* is used as the supervised learning target. In the preprocessing stage prior to training, z-score normalization was applied to the input features. The dataset was finally divided into training and evaluation subsets to assess stationary-bias regression performance. The dataset configuration is summarized in Table 1.

Table 1: Summary of dataset configuration.

Item	Description
Sensor	Chest-mounted 6-axis IMU
Measured signals	3-axis angular velocity, 3-axis accelerometer output, orientation
Time information	Timestamp and elapsed-time information recorded synchronously
Acquisition protocol	Segment-based recording with predefined motion types and durations
Motion types	Stationary, straight walking, in-place turning, walk–stop–walk transitions, small body movements
Learning target	Stationary-window-based bias proxy b_m^*
Model input	Windowed sequences of ω_z and dt
Preprocessing	Z-score normalization
Data split	Training/evaluation subsets
Objective	Stationary bias learning and heading drift compensation in PDR

3.2 Overall Framework

The proposed system estimates the two-dimensional pedestrian position using accelerometer and gyroscope data collected from a chest-mounted 6-axis IMU. As shown in Fig. 1, the pipeline consists of step detection, step length estimation, stationary window construction, DBN-based stationary bias estimation, bias-corrected heading propagation, and position update.

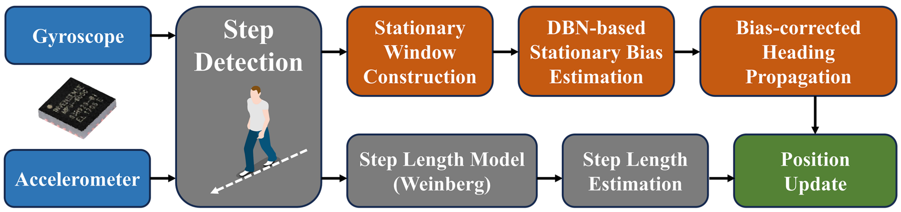

Figure 1: Overview of the proposed stationary-window-based DBN PDR framework. Accelerometer data are used for step estimation, while gyroscope data are used for stationary-window construction, DBN-based bias estimation, and bias-corrected heading propagation.

Accelerometer measurements are used for step detection and step length estimation, while gyroscope measurements are used for stationary-window construction and bias-corrected heading integration. The corrected heading is then combined with step length to obtain the final position.

The proposed framework estimates the bias pattern by leveraging the gyroscope response observed during stationary intervals and incorporates it into the heading integration during subsequent walking periods, thereby mitigating the accumulation of long-term heading drift. This enables more stable heading propagation by utilizing bias information learned from stationary intervals while preserving the physical position update mechanism of conventional PDR.

3.2.1 Step Detection

The acceleration data are first processed using a low-pass filter to suppress high-frequency noise. The magnitude of the acceleration vector at time step k is computed as:

$$a_{\text{norm},k} = \sqrt{a_{x,k}^2 + a_{y,k}^2 + a_{z,k}^2}, \tag{1}$$

where $a_{x,k}$, $a_{y,k}$, and $a_{z,k}$ denote the acceleration components along the x-, y-, and z-axes, respectively.

Step detection is performed from the temporal variation of $a_{\text{norm},k}$ using a hysteresis mechanism with an upper threshold T_{on} and a lower threshold T_{off}. A step interval starts when $a_{\text{norm},k}$ exceeds T_{on} and ends when it falls below T_{off}. Within each interval, the maximum and minimum acceleration magnitudes are recorded as $a_{\max,k}$ and $a_{\min,k}$, respectively, reducing duplicate detections caused by sensor noise.

3.2.2 Step Length Estimation

For each detected step, the step length is estimated using the Weinberg method [20]. Let SL_k denote the step length of the k-th step, defined as:

$$SL_k = K \cdot (a_{\max,k} - a_{\min,k})^{1/4}, \tag{2}$$

where K is a scaling constant determined by walking characteristics and experimental conditions. In this study, $K = 0.43$ is used as it provides empirically stable performance.

3.2.3 Stationary Window Construction

Since gyroscope bias can be most reliably estimated during stationary intervals, this section defines the criteria for constructing stationary windows. To this end, a stationary indicator is first defined to determine whether each time step corresponds to a stationary state. Let s_k denote the stationary indicator at time step k, defined as:

$$s_k = \begin{cases} 1, & \text{if } |a_{\text{norm},k} - g| < \tau_a \text{ and } |\omega_{z,k}| < \tau_\omega, \\ 0, & \text{otherwise,} \end{cases} \tag{3}$$

where g denotes the gravitational acceleration, and τ_a and τ_ω represent the thresholds for acceleration-based and angular velocity-based stationary detection, respectively. In other words, a sample is considered stationary when the acceleration norm is close to the gravitational value and the z-axis angular velocity is sufficiently small.

Based on the defined stationary indicator, a stationary window is constructed when consecutive stationary samples persist for a sufficient duration. Specifically, the m-th stationary window $\mathcal{W}_m$ is defined as:

$$\mathcal{W}_m = \{k - L + 1, \ldots, k\}, \quad \text{if } s_i = 1, \ \forall i \in \mathcal{W}_m, \tag{4}$$

where L denotes the minimum number of consecutive stationary samples required to form a valid stationary window. When the stationary condition is violated, the current window is terminated, and bias estimation is updated only when a new stationary window is formed.

3.2.4 DBN-Based Stationary Bias Estimation

Under ideal stationary conditions, the z-axis gyroscope output should be zero. Therefore, the mean response observed within a stationary interval can be regarded as a proxy for the z-axis gyroscope bias. For the m-th stationary window, the reference bias proxy b_m^* is defined as:

$$b_m^* = \frac{1}{|\mathcal{W}_m|} \sum_{i \in \mathcal{W}_m} \omega_{z,i}, \tag{5}$$

where $\omega_{z,i}$ denotes the z-axis angular velocity measured at the i-th sample within $\mathcal{W}_m$.

The DBN input consists of sequential measurements collected within the stationary window:

$$\mathbf{x}_m = [\omega_{z,k-L+1}, \ldots, \omega_{z,k}, dt_{k-L+1}, \ldots, dt_k]^\top. \tag{6}$$

The DBN estimates the stationary-window bias as:

$$\hat{b}_m = f_{\text{DBN}}(\mathbf{x}_m), \tag{7}$$

where $\hat{b}_m$ denotes the estimated z-axis gyroscope bias for the m-th stationary window.

In the proposed formulation, the stationary-window mean in Eq. (5) serves as an analytical supervision anchor for stationary gyroscope-bias learning. The DBN receives the complete temporal sequence of z-axis angular-velocity and sampling-interval values in Eq. (6) and learns a window-level mapping to the scalar bias estimate in Eq. (7). This sequence-based formulation enables the estimator to represent bias-related IMU responses observed across stationary windows and to provide an adaptive correction term for subsequent heading propagation.

This estimation mechanism establishes a clear distinction from the Window Mean baseline in Eqs. (17) and (18). The Window Mean method directly applies the per-window analytical mean to the following walking segment, whereas the DBN-based estimator learns a stationary-window representation from temporal IMU sequences collected across the training set. Through this learned representation, the proposed framework integrates stationary bias estimation with online heading correction in a unified PDR pipeline.

3.2.5 Bias-Corrected Heading Propagation

The heading is accumulated using the z-axis gyroscope measurement along with the most recently updated bias estimate. During stationary periods, the heading is maintained while the bias may be updated:

$$\theta_k = \theta_{k-1}, \quad \text{if } s_k = 1. \tag{8}$$

During motion periods, the measured angular velocity is corrected using the most recent stationary-window bias estimate $\hat{b}_m$:

$$\theta_k = \theta_{k-1} + (\omega_{z,k} - \hat{b}_m)dt_k, \quad \text{if } s_k = 0, \tag{9}$$

where θ_k denotes the accumulated heading at time step k, and dt_k represents the sampling interval. This formulation integrates the rotational motion after removing the bias component learned from stationary intervals.

3.2.6 Position Update

Once a step is completed, the current position is updated using the step length and accumulated heading. To associate each step displacement with the representative walking direction, the heading at the time of peak acceleration within the detected step interval is stored as θ_p and used for position update. The two-dimensional position (x_k, y_k) is computed as:

$$\begin{aligned} x_k &= x_{k-1} + SL_k \cos(\theta_p), \\ y_k &= y_{k-1} + SL_k \sin(\theta_p), \end{aligned} \tag{10}$$

where θ_p denotes the heading at the peak acceleration within the step interval, and $\cos(\theta_p)$ and $\sin(\theta_p)$ project the step length onto the x- and y-axes, respectively.

Algorithm 1 summarizes the finite online procedure. At each sample, the stationary state is determined, the DBN bias is updated only when a valid stationary window is formed, and the most recent bias estimate is used for heading propagation during motion. The position is updated when a step completion is detected.

3.3 Proposed Stationary-Window-Based DBN Framework

The DBN is configured as a stationary-window-level regression model for estimating the z-axis gyroscope bias. The choice of a DBN is motivated by the structure of the proposed estimation problem and by its distinction from the comparison methods. The Window Mean baseline directly applies an analytical per-window mean, whereas LSTM and GRU rely on recurrent hidden-state updates to model temporal dependencies over sequential inputs. In contrast, the proposed task is a window-level bias-representation problem that maps a short stationary-window IMU sequence $\mathbf{x}_m$ extracted from $\mathcal{W}_m$ to a scalar gyroscope-bias estimate $\hat{b}_m$ for subsequent heading correction, rather than a long-horizon trajectory prediction problem.

This setting favors a compact hierarchical representation learner that summarizes stationary-window gyroscope response patterns while preserving the physical step-wise PDR update mechanism. The stacked RBM-based architecture enables layer-wise encoding of stationary-window responses, and the feed-forward regression output provides an efficient bias estimate $\hat{b}_m$ for online heading propagation.

Algorithm 1: PDR with stationary-window-based DBN bias update

Require: IMU sequence $\{a_x[k], a_y[k], a_z[k], \omega_z[k], dt[k]\}_{k=1}^{N}$
Ensure: Estimated trajectory $\{x[k], y[k]\}_{k=1}^{N}$

Initialize $x[0] \leftarrow 0$, $y[0] \leftarrow 0$, $\theta_0 \leftarrow 0$, $\hat{b} \leftarrow 0$, $\mathcal{B} \leftarrow \varnothing$
Set $T_{\text{on}}, T_{\text{off}}, \tau_a, \tau_\omega, L, K$
for $k = 1$ to N **do**
    $a_{\text{norm},k} \leftarrow \sqrt{a_x[k]^2 + a_y[k]^2 + a_z[k]^2}$
    Set $s_k \leftarrow 1$ if $|a_{\text{norm},k} - g| < \tau_a$ and $|\omega_z[k]| < \tau_\omega$; otherwise set $s_k \leftarrow 0$
    if $s_k = 1$ **then**
        $\theta_k \leftarrow \theta_{k-1}$; append $(\omega_z[k], dt[k])$ to $\mathcal{B}$
        if $|\mathcal{B}| = L$ **then**
            $\hat{b} \leftarrow f_{\text{DBN}}(\mathbf{x}_m(\mathcal{B}))$; $\mathcal{B} \leftarrow \varnothing$
        end if
    else
        $\mathcal{B} \leftarrow \varnothing$; $\theta_k \leftarrow \theta_{k-1} + (\omega_z[k] - \hat{b})dt[k]$
    end if
    Update step state and track $a_{\max}$, $a_{\min}$, and θ_p
    if a step is completed at k **then**
        $SL_k \leftarrow K(a_{\max} - a_{\min})^{1/4}$
        $x[k] \leftarrow x[k-1] + SL_k \cos(\theta_p)$; $y[k] \leftarrow y[k-1] + SL_k \sin(\theta_p)$
    else
        $x[k] \leftarrow x[k-1]$; $y[k] \leftarrow y[k-1]$
    end if
end for
return $\{x[k], y[k]\}_{k=1}^{N}$

The overall architecture of the proposed DBN model is illustrated in Fig. 2. The model consists of six hidden layers with a structure of 50–25–50–25–50–25 units. Each hidden layer uses sigmoid units, and the hidden representations are constructed through stacked Restricted Boltzmann Machines (RBMs) during layer-wise pretraining. After the stacked RBM pretraining stage, a linear regression output layer is attached to estimate the stationary gyroscope bias $\hat{b}_m$.

In the DBN construction stage, each pair of consecutive layers is regarded as an RBM. The visible layer of the first RBM corresponds to the input vector $\mathbf{x}_m$, and the hidden representation learned by each RBM is used as the visible input to the next RBM. This stacked RBM structure can be expressed as:

$$\text{RBM}^{(l)} : \mathbf{h}^{(l-1)} \leftrightarrow \mathbf{h}^{(l)}, \quad l = 1, \ldots, L_h, \tag{11}$$

where $\mathbf{h}^{(0)} = \mathbf{x}_m$ and L_h denotes the number of hidden layers. The bidirectional connections in Fig. 2 indicate the RBM-based layer-wise pretraining stage.

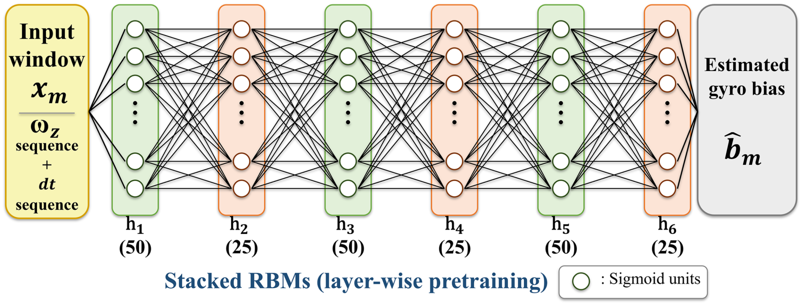

Figure 2: Architecture of the proposed DBN-based stationary gyroscope bias estimation model. The input $\mathbf{x}_m$ consists of windowed ω_z and dt sequences, and the stacked Restricted Boltzmann Machines (RBMs) with sigmoid units produce the scalar bias estimate $\hat{b}_m$.

After layer-wise pretraining, the DBN is used as a directed feed-forward regression network for supervised fine-tuning and inference. The output of the l-th hidden layer $\mathbf{h}^{(l)}$ is computed as:

$$\mathbf{h}^{(l)} = \sigma\left(\mathbf{W}^{(l)}\mathbf{h}^{(l-1)} + \mathbf{b}^{(l)}\right), \tag{12}$$

where $\sigma(\cdot)$ denotes the sigmoid activation function, and $\mathbf{W}^{(l)}$ and $\mathbf{b}^{(l)}$ represent the weight matrix and bias vector of the l-th hidden layer, respectively.

At the output layer, the stationary bias estimate $\hat{b}_m$ is computed using a linear regression layer:

$$\hat{b}_m = \mathbf{W}^{(o)}\mathbf{h}^{(L_h)} + b^{(o)}, \tag{13}$$

where $\mathbf{W}^{(o)}$ and $b^{(o)}$ denote the output-layer weight and bias, respectively. The output layer is not treated as an additional hidden layer; it only maps the final hidden representation $\mathbf{h}^{(L_h)}$ to the scalar bias estimate $\hat{b}_m$.

The model is trained to minimize the mean squared error (MSE) between the predicted bias $\hat{b}_m$ and the reference bias proxy b^*_m. The loss function for M stationary-window samples is defined as:

$$\mathcal{L} = \frac{1}{M}\sum_{m=1}^{M}\left(\hat{b}_m - b^*_m\right)^2. \tag{14}$$

Here, b^*_m is the reference bias proxy computed from the stationary window $\mathcal{W}_m$ in Eq. (5) and serves as the analytical supervision anchor for training the DBN mapping in Eq. (7).

The notations used in this study are summarized in Table 2, and the DBN model structure and training parameters are presented in Table 3.

Stochastic gradient descent (SGD) was employed for supervised fine-tuning, with a learning rate of 0.01 and 500 training epochs. The learning rate was empirically selected to provide stable convergence, and 500 epochs were used to ensure sufficient minimization of the training loss.

Prediction performance was evaluated using error-based metrics. In the proposed framework, the DBN estimates the gyroscope bias at the stationary-window level, and the estimated bias is subsequently used in the heading integration process during walking intervals to mitigate the accumulation of long-term drift.

Table 2: Notation used in the proposed method.

Symbol	Description
$a_{x,k}, a_{y,k}, a_{z,k}$	Acceleration components at time step k
$a_{\text{norm},k}$	Magnitude of acceleration vector
$\omega_{z,k}$	z-axis angular velocity
dt_k	Sampling interval
θ_k	Accumulated heading
θ_p	Heading at peak acceleration within a step
x_k, y_k	Estimated 2D position
SL_k	Estimated step length
s_k	Stationary indicator
$\mathcal{W}_m$	m-th stationary window
$\mathbf{x}_m$	DBN input feature vector
b_m^*	Reference bias proxy
$\hat{b}_m$	Estimated bias

Table 3: DBN model configuration and training parameters.

Parameter	Value
Model type	Stacked RBM-based DBN regression model
Input feature	Flattened ω_z and dt sequences from $\mathcal{W}_m$
Number of hidden layers L_h	6
Hidden layer structure	50–25–50–25–50–25
Activation function	Sigmoid
Output layer	Linear regression layer for scalar $\hat{b}_m$
Loss function	Mean squared error (MSE)
Fine-tuning optimizer	Stochastic gradient descent (SGD)
Learning rate	0.01
Training epochs	500

4 Evaluation

4.1 Experimental Settings

This section evaluates the proposed stationary-window-based DBN bias update under various indoor walking conditions. The evaluation was conducted under three scenarios: a short-term closed-loop path, a repeated long-term path, and a complex multi-turn path. In each experiment, the ground-truth waypoints were predefined from the measured path geometry, and the waypoint-based absolute trajectory error (ATE) was computed as the Euclidean distance between each ground-truth waypoint and its corresponding estimated point selected from the detected turning events and endpoint.

The waypoint-based ATE used in this study is not based on full trajectory alignment but is defined as the Euclidean distance between the ground-truth and estimated positions at each waypoint. This evaluation not only measures the final endpoint error but also captures how heading drift accumulates at intermediate turning points and corners, and how effectively the proposed bias update mitigates such accumulated errors.

For N_{wp} evaluated waypoints, the waypoint-based error and its summary metrics are defined as:

$$\begin{aligned}
e_i &= \left[(x_i^{\mathrm{pred}} - x_i^{\mathrm{gt}})^2 + (y_i^{\mathrm{pred}} - y_i^{\mathrm{gt}})^2\right]^{1/2}, \\
\mathrm{ATE}_{\min} &= \min_{1\le i\le N_{\mathrm{wp}}} e_i, \\
\mathrm{ATE}_{\max} &= \max_{1\le i\le N_{\mathrm{wp}}} e_i, \\
\mathrm{ATE}_{\mathrm{avg}} &= \frac{1}{N_{\mathrm{wp}}}\sum_{i=1}^{N_{\mathrm{wp}}} e_i, \\
\mathrm{ATE}_{\mathrm{RMSE}} &= \left[\frac{1}{N_{\mathrm{wp}}}\sum_{i=1}^{N_{\mathrm{wp}}} e_i^2\right]^{1/2}.
\end{aligned} \tag{15}$$

Waypoint-level variability is also summarized using the standard deviation (SD) of the waypoint-based ATE values for each method:

$$s_e = \left[\frac{1}{N_{\mathrm{wp}} - 1}\sum_{i=1}^{N_{\mathrm{wp}}}\left(e_i - \mathrm{ATE}_{\mathrm{avg}}\right)^2\right]^{1/2}, \tag{16}$$

where s_e denotes the waypoint-level SD. Since the evaluation protocol is based on predefined waypoint errors within each trajectory, the reported SD is used as a waypoint-level dispersion measure that complements the average ATE, maximum ATE, and RMSE results in Eq. (15).

The comparison models include Baseline, Window Mean, LSTM, and GRU. The Baseline denotes a Kalman-filter-based PDR method that uses the same step detection, step length estimation, and position update procedures as the proposed framework, but does not include the proposed stationary-window-based DBN bias update. Among the correction-based comparison methods, the Window Mean method estimates the bias for each stationary interval by computing the mean of the observed z-axis angular velocity. Let $\mathcal{S}_j$ denote the j-th stationary window; the bias estimate is defined as:

$$\hat{b}_j = \frac{1}{M_j}\sum_{k\in\mathcal{S}_j}\omega_{z,k}, \tag{17}$$

where $\omega_{z,k}$ denotes the z-axis angular velocity measured at the k-th sample, and M_j denotes the number of samples in the j-th stationary window. The estimated bias is then used to correct the gyroscope measurements in the subsequent walking segment, where the corrected angular velocity is given by:

$$\omega_{z,k}^{\mathrm{corr}} = \omega_{z,k} - \hat{b}_j. \tag{18}$$

Through this setup, we comparatively evaluate the direct Window Mean correction in Eqs. (17) and (18), the representative sequence models LSTM and GRU, and the proposed DBN-based stationary-window bias estimation method.

The proposed system was implemented and validated using a chest-mounted 6-axis IMU and a Raspberry Pi 4. The Raspberry Pi 4 was connected to the IMU via an Inter-Integrated Circuit (I^2C) interface and operated on Ubuntu MATE 22.04, equipped with a Broadcom BCM2711 1.5 GHz quad-core Cortex-A72 64-bit system-on-chip (SoC) and 8 GB RAM. Sensor data were collected under a lightweight wearable PDR setup.

For fair comparison, all methods were evaluated using IMU data collected under identical path conditions. The learning-based models, including DBN, LSTM, and GRU, were trained on a workstation equipped with two NVIDIA RTX 4080 graphics processing units (GPUs). The model outputs were then incorporated into the same heading integration and position update pipeline for final performance evaluation.

4.2 Experiments on Short-Term Indoor Positioning

The short-term indoor positioning experiment was conducted under a single traversal of a closed-loop path and was designed to evaluate the basic waypoint-level positioning accuracy of each method before significant drift accumulation occurs. The ground-truth trajectory consisted of five waypoints, including four turning points and the final endpoint. The waypoint-based ATE was computed as the Euclidean distance between each ground-truth waypoint and the corresponding point on the estimated trajectory. In this experiment, the total ground-truth path length was 74.48 m.

As shown in Fig. 3 and Table 4, the evaluated methods preserved the overall closed-loop trajectory under the single-lap condition. With a total ground-truth path length of 74.48 m, this experiment serves as a low-drift reference case for characterizing waypoint-level trajectory behavior in a short indoor path. The results provide a reference view of each method's trajectory response under short-path conditions, and the following long-term and complex-path experiments further examine performance under accumulated heading-drift conditions.

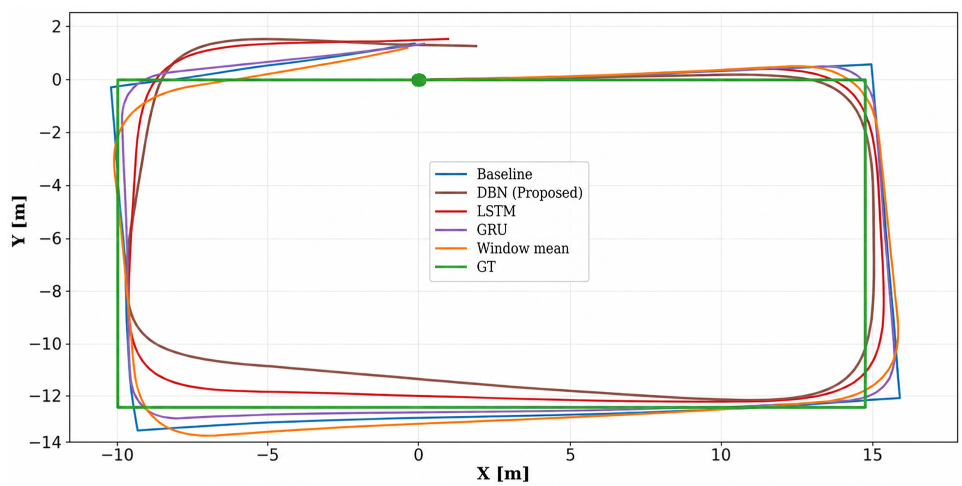

Figure 3: Trajectory comparison for the short-term closed-loop indoor positioning experiment.

Table 4: ATE statistics for the short-term experiment (unit: m).

Method	WP1	WP2	WP3	WP4	End	Avg.	SD
Baseline	0.6017	1.1949	1.1107	0.7958	1.3873	1.0181	0.3158
Window Mean	0.4036	0.9100	1.4902	1.6582	1.2654	1.1455	0.5005
GRU	0.1681	0.6275	0.8861	0.6626	1.3688	0.7426	0.4366
LSTM	0.6953	1.0129	1.6463	1.2347	1.7084	1.2595	0.4275
DBN (Proposed)	0.8737	1.1248	2.1437	1.6342	1.8950	1.5343	0.5281

4.3 Experiments on Long-Term Indoor Positioning

The long-term indoor positioning experiment was conducted by traversing the same closed-loop path three times and was designed to evaluate the robustness of each method against accumulated heading drift as the walking distance increases. As in the short-term experiment, four turning points and the final endpoint of each lap were defined as waypoints, resulting in a total of 15 waypoints over three laps. The total ground-truth path length in this experiment was 223.44 m.

As shown in Fig. 4 and Table 5, performance differences became clearer in the three-lap experiment as heading drift accumulated over repeated walking. The Window Mean, Baseline, GRU, LSTM, and proposed DBN methods showed distinct waypoint-level error patterns under the same 223.44 m path condition.

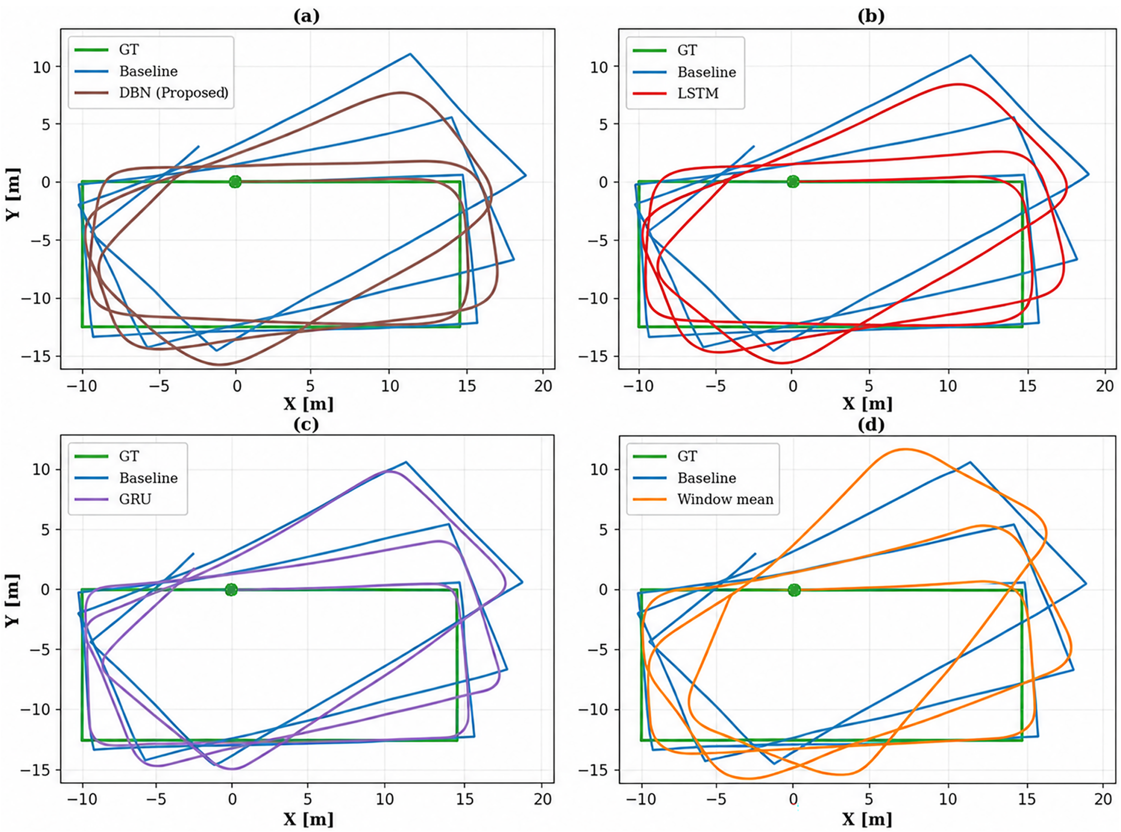

Figure 4: Trajectory comparison for the 3-lap long-term indoor positioning experiment: (**a**) the proposed DBN method, (**b**) LSTM, (**c**) GRU, and (**d**) Window mean.

Table 5: ATE statistics for the 3-lap long-term experiment (unit: m).

Method	Avg.	Min.	Max.	RMSE	SD
Baseline	4.4392	0.6017	13.3475	5.8058	3.8731
Window Mean	6.1343	0.4036	17.0293	8.0081	5.3285
GRU	4.3328	0.1681	13.3907	5.8218	4.0249
LSTM	4.1725	0.6953	11.6496	5.3181	3.4131
DBN (Proposed)	3.9587	0.8737	10.6830	4.9993	3.1603

In this accumulated-drift regime, the proposed DBN method produced the most stable waypoint-level trajectory result among the compared methods, achieving the lowest average ATE of 3.9587 m, the lowest RMSE of 4.9993 m, and the lowest waypoint-level SD of 3.1603 m. Compared with LSTM, the proposed method reduced the average ATE by approximately 5.1% and the RMSE by approximately 6.0%. These results indicate that stationary-window-based DBN bias estimation provides consistent drift-suppression behavior when heading error accumulation becomes more pronounced.

4.4 Experiments on Complex Indoor Positioning

The complex path experiment was conducted to evaluate the robustness of drift correction under conditions involving multiple direction changes and non-linear motion. Real indoor walking does not consist only of repeated straight-line motion, but also includes frequent turns, reorientation, and local motion variations. Therefore, the complex path scenario was designed to assess whether the proposed method operates reliably under realistic indoor motion variations. The total ground-truth path length in this experiment was 80.77 m, and errors were computed at six waypoints, including major turning points and the final endpoint, excluding the starting point.

As shown in Fig. 5, Tables 6 and 7, the performance gap between methods becomes more pronounced in the mid-to-late segments where multiple direction changes occur. The complex-path experiment therefore highlights the benefit of the proposed stationary-window-based estimator under frequent direction changes and turn-intensive indoor motion.

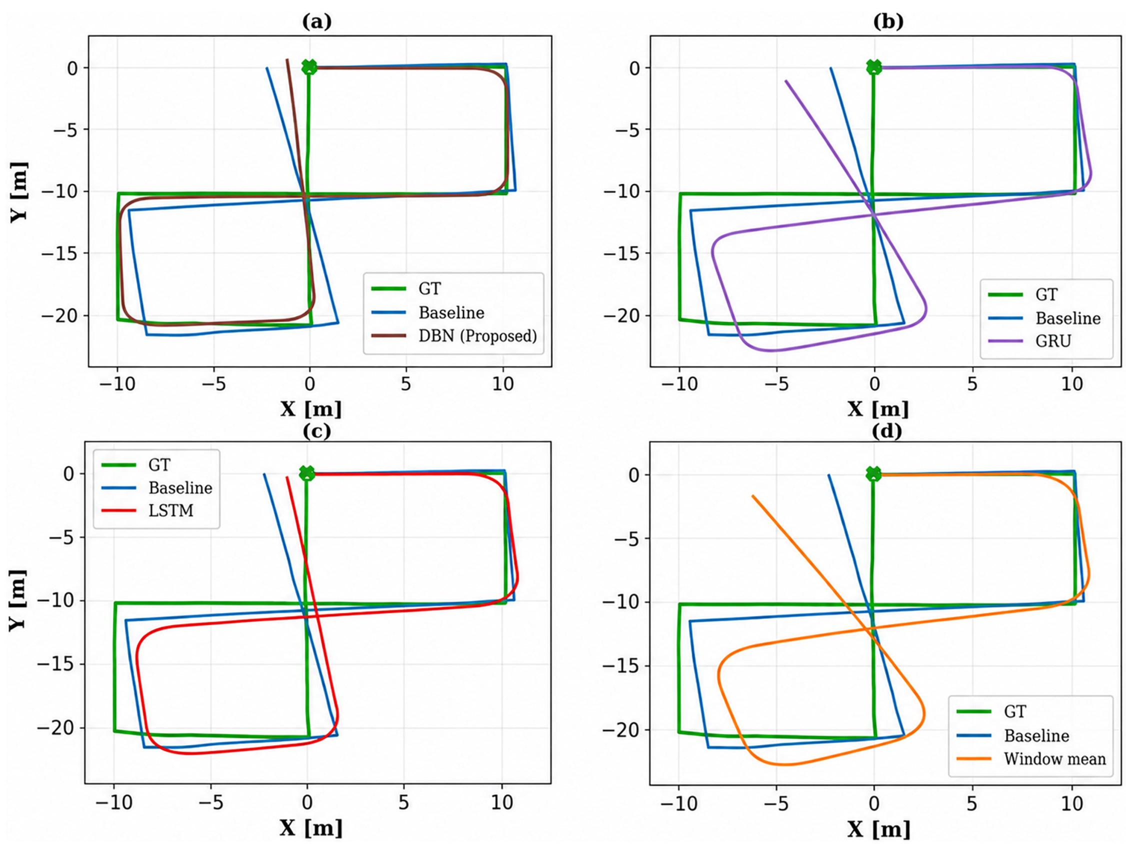

Figure 5: Trajectory comparison for the complex indoor positioning experiment: (**a**) the proposed DBN method, (**b**) GRU, (**c**) LSTM, and (**d**) Window mean.

Table 6: Waypoint ATE results for the complex path experiment (unit: m).

Method	WP1	WP2	WP3	WP4	WP5	End
Baseline	0.2131	0.5082	1.3709	1.9322	1.4709	2.1875
Window Mean	1.0103	0.4742	4.6751	5.1541	2.6200	6.3457
GRU	0.6763	0.3694	3.9315	4.5605	2.5597	4.6388
LSTM	0.9051	0.3460	2.7627	2.6750	1.1054	1.0784
DBN (Proposed)	0.9041	0.7280	0.8973	0.7474	0.9935	1.2052

Table 7: ATE statistics for the complex path experiment (unit: m).

Method	Avg.	Min.	Max.	RMSE	SD
Baseline	1.2805	0.2131	2.1875	1.4643	0.7781
Window Mean	3.3799	0.4742	6.3457	4.0166	2.3773
GRU	2.7894	0.3694	4.6388	3.2894	1.9098
LSTM	1.4787	0.3460	2.7627	1.7374	0.9991
DBN (Proposed)	0.9126	0.7280	1.2052	0.9265	0.1754

Quantitatively, the proposed DBN method achieved the lowest average ATE of 0.9126 m, the lowest RMSE of 0.9265 m, the lowest maximum ATE of 1.2052 m, and the lowest waypoint-level SD of 0.1754 m, corresponding to approximately 1.13%, 1.15%, and 1.49% of the total path length for the average ATE, RMSE, and maximum ATE, respectively. The proposed method also maintained a compact waypoint-error range from 0.7280 to 1.2052 m across the six evaluated waypoints. Compared with LSTM, the proposed DBN method reduced the average ATE by approximately 38.3% and the RMSE by approximately 46.7%.

These results show that the learned stationary-window bias representation contributes to stable heading propagation in complex multi-turn paths. The compact waypoint-error range and the low waypoint-level SD of the proposed method further indicate consistent trajectory behavior across the entire complex indoor path.

4.5 Stationary Detection Error Sensitivity

Finally, a controlled sensitivity experiment was conducted on the same complex path to examine the effect of stationary-decision outcomes on the proposed bias update. Four cases were compared: Reference, Actual detector, Injected FP 10%, and Injected FN 10%. The ATE statistics were computed using the same waypoint and endpoint correspondence criterion as in the complex path experiment.

As shown in Fig. 6 and Table 8, the Actual detector case produced a trajectory closely aligned with the Reference case. Quantitatively, the Actual detector achieved an average ATE of 0.6999 m and an RMSE of 0.9096 m, which are close to the Reference values of 0.7121 and 0.9202 m, respectively. This indicates that the implemented stationary detector provides sufficiently valid stationary windows for DBN-based bias estimation.

In contrast, Injected FP 10% caused a substantial increase in positioning error, increasing the average ATE and RMSE to 6.2238 and 7.2009 m, respectively. This occurs because walking or turning segments can be incorrectly treated as stationary, causing true rotational motion to be removed as bias. Injected FN 10% also degraded performance, but its impact was smaller, with an average ATE of 2.7123 m and an RMSE of

3.2685 m. These results show that false positives are more critical than false negatives for stationary-window-based bias update.

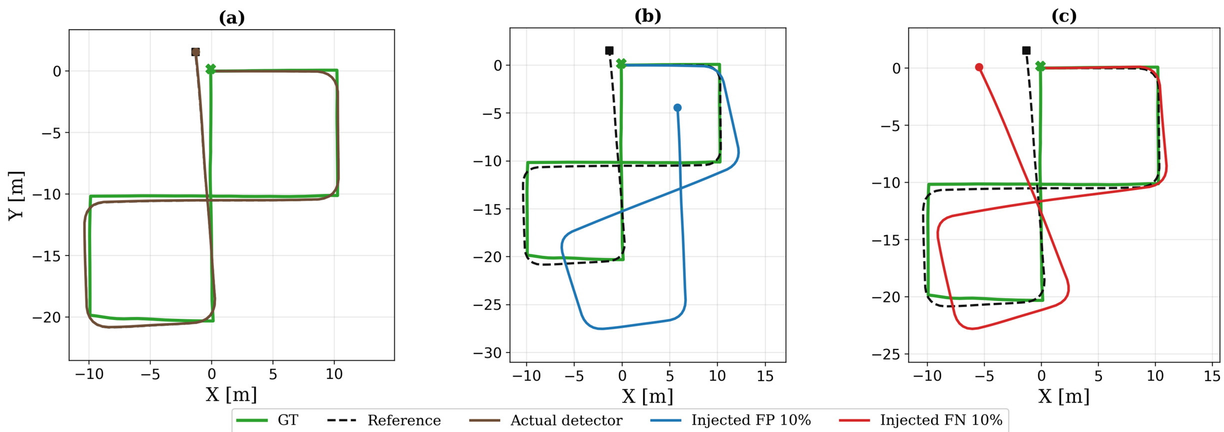

Figure 6: Trajectory comparison for stationary detection sensitivity: (**a**) Actual detector, (**b**) Injected FP 10%, and (**c**) Injected FN 10%.

Table 8: Waypoint-based stationary detection error sensitivity analysis (unit: m).

Case	Avg. ATE	Min. ATE	Max. ATE	RMSE	Endpoint Error
Reference	0.7121	0.1420	1.8513	0.9202	1.8513
Actual detector	0.6999	0.1357	1.8407	0.9096	1.8407
Injected FP 10%	6.2238	0.2433	9.9353	7.2009	7.4806
Injected FN 10%	2.7123	0.0812	5.3935	3.2685	5.3935

5 Conclusion

This paper proposed a stationary-window-based DBN framework for heading-drift correction in IMU-based PDR. The proposed method learns a gyroscope bias representation from z-axis angular-velocity and sampling-interval sequences observed during stationary intervals and applies it as a correction term during subsequent heading integration. By updating the correction term when a valid stationary window is detected, the proposed framework preserves the conventional PDR structure while supporting stable trajectory estimation in long-duration and complex-path scenarios.

The experimental results show that the proposed method is particularly effective under long-term and complex walking conditions where heading drift accumulation becomes more pronounced. In the three-lap long-term experiment, the proposed DBN method achieved the lowest average ATE of 3.9587 m and the lowest RMSE of 4.9993 m. In the complex path experiment, the proposed DBN method also achieved the lowest overall error, with an average ATE of 0.9126 m and an RMSE of 0.9265 m. These results show that stationary-window-based DBN bias estimation provides stable waypoint-level trajectory behavior in accumulated-drift regimes.

Future work will further extend the proposed framework by improving stationary detection robustness, incorporating additional contextual information such as sensor orientation, environmental variation, and long-term temporal characteristics, and validating the method under more diverse users, sensor placements, and indoor path configurations. These extensions will support broader deployment of stationary-window-based bias learning for standalone wearable PDR systems.

Acknowledgement: The authors appreciate the members of the WINE Laboratory at Korea University for their helpful discussions and technical assistance. During revision, AI-assisted tools were used only for language polishing, structural adjustment, and LaTeX formatting support. All technical content, experiments, numerical results, and conclusions were reviewed and approved by the authors.

Funding Statement: This research was supported in part by the IITP (Institute of Information & Communications Technology Planning & Evaluation)-ITRC (Information Technology Research Center) grant funded by the Korea government (Ministry of Science and ICT) (IITP-2026-RS-2021-II211835), and in part by the Korea Institute of Energy Technology Evaluation and Planning (KETEP) and the Ministry of Climate, Energy Environment (MCEE) of the Republic of Korea (No. RS-2022-KP002860).

Author Contributions: Conceptualization, methodology, validation, and writing, Kyeonghyun Yoo, Sangmin Lee, and Hwangnam Kim; software, Kyeonghyun Yoo; formal analysis, Kyeonghyun Yoo and Hwangnam Kim; investigation, Kyeonghyun Yoo and Sangmin Lee; supervision, project administration, and funding acquisition, Hwangnam Kim. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The data used in this study are not owned by the authors and cannot be made publicly available. Access is restricted by the data owner in accordance with project policies.

Ethics Approval: Not applicable. This study does not involve human participants or animal subjects.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Harle R. A survey of indoor inertial positioning systems for pedestrians. IEEE Commun Surv Tutor. 2013;15(3):1281–93. doi:10.1109/surv.2012.121912.00075.
2. Chen C, Lu CX, Wahlström J, Markham A, Trigoni N. Deep neural network based inertial odometry using low-cost inertial measurement units. IEEE Trans Mob Comput. 2021;20(4):1351–64. doi:10.1109/tmc.2019.2960780.
3. Chen C, Zhao P, Lu CX, Wang W, Markham A, Trigoni N. Deep-learning-based pedestrian inertial navigation: methods, data set, and on-device inference. IEEE Internet Things J. 2020;7(5):4431–41. doi:10.1109/jiot.2020.2966773.
4. Lee S, Yoo S, Lee JY, Park S, Kim H. Drone positioning system using UWB sensing and out-of-band control. IEEE Sens J. 2022;22(6):5329–43. doi:10.1109/jsen.2021.3127233.
5. Tong X, Su Y, Li Z, Si C, Han G, Ning J, et al. A double-step unscented Kalman filter and HMM-based zero-velocity update for pedestrian dead reckoning using MEMS sensors. IEEE Trans Ind Electron. 2020;67(1):581–91. doi:10.1109/tie.2019.2897550.
6. Li H, Guo H, Qi Y, Deng L, Yu M. Research on multi-sensor pedestrian dead reckoning method with UKF algorithm. Measurement. 2021;169:108524.
7. Qiu S, Wang Z, Zhao H, Qin K, Li Z, Hu H. Inertial/magnetic sensors based pedestrian dead reckoning by means of multi-sensor fusion. Inf Fusion. 2018;39(1):108–19. doi:10.1016/j.inffus.2017.04.006.
8. Muthineni K, Artemenko A, Vidal J, Nájar M, Catalan M, Paradells J. Deep Learning-Based UWB-IMU data fusion for indoor positioning in industrial scenario. IEEE Open J Veh Technol. 2025;6(18):1209–21. doi:10.1109/ojvt.2025.3566888.
9. Liao W, Junna S, Wu Z, Chundong L, Mingkun S. A robust framework for PDR/magnetic fingerprint/GNSS integrated navigation with boundary constraints. Measurement. 2026;271(8):120947. doi:10.1016/j.measurement.2026.120947.

10. Chen LW, Huang WC. Deep learning based pedestrian dead reckoning with vision anchor augmentation for seamless centimeter-level indoor positioning. IEEE Trans Syst Man Cybern Syst. 2026;56(6):3623–35. doi:10.1109/tsmc.2026.3659913.
11. Liu X, Zhang Y, Li N, Huang Y, Zhao Y. A novel pedestrian dead-reckoning algorithm with self-estimates of IMU biases. IEEE Trans Instrum Meas. 2025;74:1–9.
12. Feigl T, Kram S, Woller P, Siddiqui RH, Philippsen M, Mutschler C. A bidirectional LSTM for estimating dynamic human velocities from a single IMU. In: Proceedings of the 2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN); 2019 Sep 30–Oct 3; Pisa, Italy. p. 1–8.
13. Hua Y, Guo J, Zhao H. Deep belief networks and deep learning. In: Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things; 2015 Oct 8–10; Delhi, India. p. 1–4.
14. Kone Y, Zhu N, Renaudin V, Ortiz M. Machine learning-based zero-velocity detection for inertial pedestrian navigation. IEEE Sens J. 2020;20(20):12343–53. doi:10.1109/jsen.2020.2999863.
15. Lee D, Yoo K, Jung W, Kim H. GRUI: a novel gesture recognition utilizing UWB sensor and IMU. In: Proceedings of the 2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC); 2024 Oct 6–10; Kuching, Malaysia. p. 1163–8.
16. Li X, Shen W, Gu P, Zou J, Lyu X, Meng F, et al. Learning-based motion-aware dead reckoning using IMU and wheel encoders. IEEE Trans Instrum Meas. 2026;75:1–14. doi:10.1109/tim.2026.3662904.
17. Pham TT, Suh YS. Conditional generative adversarial network-based regression approach for walking distance estimation using waist-mounted inertial sensors. IEEE Trans Instrum Meas. 2022;71(2):1–13. doi:10.1109/tim.2022.3177730.
18. Liu W, Caruso D, Ilg E, Dong J, Mourikis AI, Daniilidis K, et al. TLIO: tight learned inertial odometry. IEEE Robot Autom Lett. 2020;5(4):5653–60.
19. Herath S, Yan H, Furukawa Y. RoNIN: robust neural inertial navigation in the wild: benchmark, evaluations, & new methods. In: Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA); 2020 May 31–Aug 31. Paris, France. p. 3146–52.
20. Weinberg H. Using the ADXL202 in pedometer and personal navigation applications. Analog Devices. 2002;2(2):1–6.