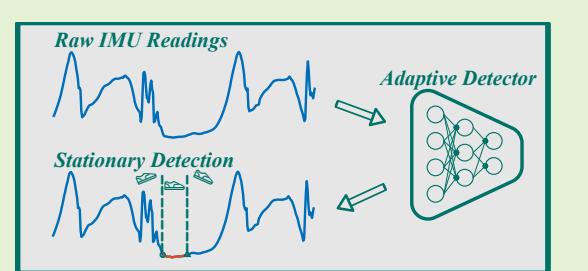


Symmetrical-Net: Adaptive Zero Velocity Detection for ZUPT-Aided Pedestrian Navigation System

Mingkun Yang, Ran Zhu, Zhuoling Xiao, Bo Yan

Abstract—Inertial navigation system (INS) is a practical method for indoor pedestrian navigation without pre-installation of infrastructure. Based on the fundamentals of human bipedal motion, zero velocity update (ZUPT) is a pervasive approach to tackle the accumulated error of inertial measurement units (IMU). While zero velocity detection plays a vital role in the algorithm, existing fixed-threshold methods to pick these pseudo-measurements of error-state Kalman Filter (ESKF) have the doubtful capability to fit various individuals in different motions. To address this issue, we propose the Symmetrical-Net leveraging deep Recurrent Convolutional Neural Networks (RCNNs) to detect the zero velocity interval adaptively. Additionally, two RCNNs are constructed in the symmetrical framework considering bidirectional IMU readings, which further improves the performance of the model. A comprehensive evaluation containing 87 different trajectories from 27 individuals has been conducted. The results show that the detection accuracy is up to 99.5% and 96.5% on the training and validation sets, respectively. It is verified that the precise and robust Symmetrical-Net can be a viable approach for the ZUPT-aided INS system.

Index Terms—Inertial measurement units, Zero velocity update, Pedestrian dead reckoning, Machine learning



I. INTRODUCTION

INDOOR pedestrian tracking cannot count on Global Positioning System (GPS) owing to its poor penetration. Therefore, researchers have proposed various radio frequency localization systems (i.e., Wi-Fi, Bluetooth, ibeacon, etc.) for the indoor environment [1]–[3], but all require installation of infrastructure beforehand. The burden of deploying beacons or landmarks falls on such systems, which is unrealistic in some conditions, especially in the rescue scene [4]–[6].

Pedestrian dead reckoning (PDR) with inertial measurement units (IMU), as the self-contained navigation system without the requirement for ambient conditions, is a justifiable solution for general indoor localization. Nevertheless, the inertial navigation system (INS) calculates the relative displacement of an object that will result in accumulated errors due to the sensors' drift. Stable trajectories could only maintain for a very short term using commercial IMU. To address this issue, foot-mounted IMU with zero velocity update (ZUPT) was proposed to eliminate the divergence of trajectory over time [7]. The argument about attaching IMU to feet is that bipedal gait can split into two phases: the stance phase and the swing phase.

This work was supported by National Natural Science Foundation of China Grant No. 61703076 and No. 61973056.

The authors are with the Department of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China (e-mail: {mingkunyang, ranzhu}@std.uestc.edu.cn, zhuolingxiao@gmail.com, yanboyu@uestc.edu.cn).

Under the assumption that the velocity is zero when the foot is stationary, it provides error-state Kalman Filter (ESKF), the drift corrector, pseudo-measurements to modify the position error [8].

The key to ZUPT with ESKF is precise zero velocity detection that distinguishes the stationary phase from each stride. Due to the complex differences in pedestrian motion patterns, the hope is that the zero velocity detector becomes robust for various individuals. While using other sensors besides IMU [9]–[11] can realize precise detection, these systems tend to be intricate because of synchronizing each module. Utilizing the raw IMU readings only, we present a novel approach leveraging deep learning (DL) to detect zero velocity adaptively. Trained by massive foot-mounted IMU data from different individuals, the symmetrical Recurrent Convolutional Neural Network (RCNNs) can effectively learn the law of gait because the model takes the information from forward to backward of the undetermined time instant into consideration. The Convolutional Neural Network (CNN) and the Long Short-Term Memory (LSTM) in Symmetrical-Net extracts features and contextual information from input sequences, respectively. The proposed adaptive approach, unlike the conventional fixed threshold detector, requires no tuning on the optimal threshold, which will significantly elevate the applicability of ZUPT-aided INS. In summary, the main contributions of this paper are as follows:

- Robust and adaptive zero velocity detection: Leverag-

ing the neural network, more robust and adaptive zero velocity detection is realized compared with the fixed threshold. The symmetrical framework also guarantees precise detection regardless of motion types and individual differences.

- Accurate and robust pedestrian tracking: Performance of zero velocity detector makes a profound impact on the trajectory reconstruction of the ZUPT-aided INS system. The navigation system utilizing our proposed zero velocity detection shows significant improvement in terms of tracking accuracy and robustness.
- Extensive real-world validation: Huge numbers of experiments reveal that INS assisted by Symmetrical-Net keep the high tracking accuracy in multiple indoor and outdoor environments under three paces (i.e., walking, fast walking, and running). Outstanding improvements, especially in robustness, on zero velocity detection via the proposed model are corroborated.

The remainder of this paper is organized as follows: Section II overviews the related work focusing on zero velocity detection methods for ZUPT-aided PDR. In Section III, we give an exhaustive discussion of the Symmetrical-Net. Abundant evaluation will be presented in Section IV, and Section V concludes the paper.

II. RELATED WORK

After the foot-mounted IMU with ZUPT-aided pedestrian navigation system, known as Navshoe [7], was proposed for GPS-denied environments, researches based on this state-of-the-art model has made remarkable progress [12]. Among them, zero velocity detection is of vital importance in the entire ZUPT with ESKF drift elimination module. During pedestrian moving, two feet are in the periodic process. Each gait cycle consists of almost 60% stance phase and 40% swing phase, as shown in Fig. 1. The stance phase is defined as the period when the foot is on the ground, and the swing phase refers to the period when the foot is in the air [13], [14]. In terms of the zero velocity we select, it is the interval that the whole foot is anchored to the ground [15], [16]. In this paper, the stationary phase is defined as the specific stance phase that contains these zero velocity points. Note that the stationary phase lasts from foot flat to heel off corresponding to the foot conditions in Fig. 1. We focus on the techniques that aim to improve the accuracy of classifying the zero velocity point, which can be divided into three classes: 1) Sensors fusion techniques that make use of other sensors besides IMU. 2) Likelihood ratio test (LRT) techniques that design hypothesis-test methods. 3) Machine learning techniques, the data-driven approaches, output the probability, whether it is stationary for each sampling raw IMU data.

A. Sensors Fusion

Sensor fusion adds observations to the INS to assist zero velocity detector. Bebek et al. [9] used a pressure sensor attached to the heel as biomechanical ground reaction sensors cluster (GRSC) to select zero velocity points accurately. Skog et al. [10] constructed a zero velocity detector by mounting

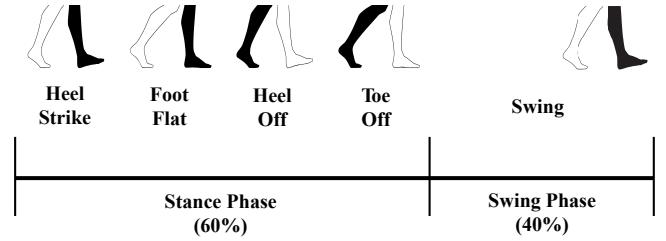


Fig. 1. One gait cycle of the right foot (black). The entire cycle can be divided into two phases: the stance phase (occupies 60%) and the swing phase (occupies 40%).

three force sensitive resistors (FSRs) between the midsole and outsole of the boot. The measured resistances provided a measure of the forces put to the ground. Moreover, dual-foot systems [11], [17], which merged two IMU placed on pedestrians left and right foot respectively, provided a reliable and periodic equality constraint. It is indeed an effective way to enhance classification accuracy by combining additional measurements. However, these systems have to put effort into synchronizing all the sensors.

B. LRT with Fixed Threshold

LRT with the fixed threshold can be formalized as a hypothesis-testing problem [15], [18] using the IMU output (i.e., accelerometer or gyroscope or a mixture of the two). Acceleration magnitude (MAG) [19], acceleration moving variance (AMV) [20], angular rate energy (ARE) [21] are the common assumed prior knowledge to analyze zero velocity. Stance Hypothesis Optimal dEctector (SHOE) [10], [18] combined the aforementioned prior knowledge, which showed remarkable performance. Calculated generalized likelihood ratio (GLR) will compare with a fixed threshold derived from a large number of experiments. The generalized likelihood ratio test (GLRT) chooses the hypothesis that gait is stationary when GLR falls below the threshold. A proper threshold impacts profoundly on the detection accuracy. More intelligent systems [22], [23] leveraging the support vector machine (SVM) to classify motion based on inertial data have been proposed. Taking advantage of the outcome based on human activity recognition (HAR) [24], optimal thresholds were selected for zero velocity detector. A single threshold, typically, may display the satisfying results for a certain condition, whereas it cannot suit every individual with multiple motions. To address this problem, there have been some learning-based efforts recently that aim to dynamically detect the zero velocity intervals, as discussed in the next subsection.

C. Machine Learning Methods

Recent researches emerge that deep learning has a wide range of applications owing to the marvelous progress in computation capability. Some researchers, therefore, attempt to utilize deep learning as the scheme for zero velocity detection. Literature [25], [26] presented a method to improve the accuracy of a ZUPT-aided INS by replacing the standard zero velocity detection with the LSTM neural network. Literature [27] proposed a step detector using bidirectional LSTM, which

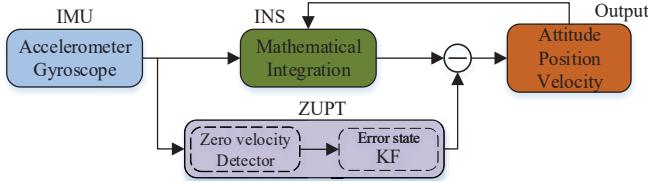


Fig. 2. Block diagram of ZUPT-aided INS. Our main contribution is conducting a deep learning algorithm, namely Symmetrical-Net, to realize adaptive zero velocity detection (in purple block).

yielded excellent accuracy for SHSs. Inspired by the above-mentioned literature, we propose a novel approach combining CNN and LSTM, assembled in a symmetrical framework. In this paper, our model achieves a higher classification accuracy compared with competing techniques.

III. SYMMETRICAL-NET

A. ZUPT-aided INS

1) Basic INS: Before describing the details of INS, we define b as the body coordinate and n as the navigation coordinate. Both two coordinates are right-handed systems. Arranged in the order of the X-Y-Z axis, the body coordinate is forward-left-up frame and the navigation coordinate is east-north-up (ENU) frame. Raw accelerometer and gyroscope readings denote a^b and ω^b , and INS output contains attitude C_b^n , velocity v^n , position p^n . The matrix C_b^n , determining the rotation from b to n , is special orthogonal group $SO(3) = \{C_b^n \in \mathbb{R}^{3 \times 3} \mid C_b^n C_b^{nT} = I, \det(C_b^n) = 1\}$, which is called direction cosine matrix (DCM) or rotation matrix.

Considering the sampling rate of sensors and pedestrian speed, we not only ignore some effects caused by earth spins, but the change of angle between two consecutive positions is small. Therefore, combined with IMU readings at k time instant, the output of INS at k time instant can be estimated by [28]:

$$C_{b_k}^n = C_{b_{k-1}}^n C_{b_k}^{b_{k-1}} \approx C_{b_{k-1}}^n (I + [\omega^{b_k}] \times T) \quad (1)$$

$$v_k^n = v_{k-1}^n + \int_T a_k^n dt \approx v_{k-1}^n + a_k^n T \quad (2)$$

$$p_k^n = p_{k-1}^n + \int_T v_k^n dt \approx p_{k-1}^n + v_{k-1}^n T + a_k^n T^2 / 2 \quad (3)$$

where $[\cdot] \times$ is the skew-symmetric matrix of vector, T is the sampling period. Considering the acceleration a^b measured by the IMU is specific force (i.e., the non-gravitational force), the acceleration in n coordinate a_k^n is calculated by $a_k^n = C_{b_k}^n a^{b_k} + g^n$ where $g^n = [0, 0, -9.81]$ compensates for the gravity projection in ENU frame.

2) ZUPT with ESKF: The architecture of a standard closed-loop ZUPT-aided INS is shown in Fig. 2. To alleviate the accumulated error, the ZUPT block based on ESKF modifies the output of INS. We define the error vector $\delta x_k = [\delta \phi_k, \delta p_k, \delta v_k]^T$ that contains the Euler angle error, position error, and velocity error. Note that the ZUPT-aided algorithm can estimate the model error caused by integrals in INS, but the situation for estimating inertial sensors error by velocity

pseudo-measurement is poor [29], [30]. Therefore, we do not update the sensors biases through ESKF.

The error dynamics is a linear system of which the transition function from $k-1$ time step to k time step is

$$\hat{\delta x}_k = F_k \delta x_{k-1} + \epsilon \quad (4)$$

where ϵ represents the randomness in the state transition, and the state transition matrix

$$F_k = \begin{bmatrix} I_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & I_{3 \times 3} & T I_{3 \times 3} \\ T[a_k^n] \times & 0_{3 \times 3} & I_{3 \times 3} \end{bmatrix}. \quad (5)$$

Because ϵ is zero mean variable, the mean of $\hat{\delta x}_k$ is given by $F_k \delta x_{k-1}$ [31]. In the prediction stage, indeed, we can not estimate the error state without external measurement. Thus, $\hat{\delta x}_k$ is a zero mean vector.

The covariance matrix of the error vector is P_k , which is in the size of 9×9 . It is updated after each INS calculation:

$$\hat{P}_k = F_k P_{k-1} F_k^T + Q \quad (6)$$

where Q is the covariance matrix of ϵ .

When the gait is in stationary phase, velocity of pedestrian is assumed as zero. Combined with the velocity calculated from INS, the pseudo-measurement of velocity error is

$$z_k = v_k^{nT} - \vec{0}. \quad (7)$$

The measurement corresponds to the velocity error δv_k in the error state $\hat{\delta x}_k$:

$$z_k = H \hat{\delta x}_k + \zeta \quad (8)$$

where ζ denotes the measurement noise, and measure matrix $H = [0_{3 \times 3}, 0_{3 \times 3}, I_{3 \times 3}]$.

The Kalman gain K_k determines the proportion of pseudo-measurement updating the error state, and it is obtained by

$$K_k = \hat{P}_k H^T (H \hat{P}_k H^T + R)^{-1} \quad (9)$$

where R is the covariance matrix of ζ . The mean of error state is estimated by

$$\delta x_k = \hat{\delta x}_k + K_k (z_k - H \hat{\delta x}_k) = K_k z_k = K_k v_k^{nT}. \quad (10)$$

The ZUPT-aided INS is summarized as

B. Zero Velocity Detection

As the vital process of ZUPT-aided INS, zero velocity detection can be defined as a binary hypothesis testing problem [10], [32], [33]. Let $x_i = [x_i^a, x_i^\omega] \in \mathbb{R}^6$ indicates IMU data of the i th sampling point, which contains the accelerometer readings $x_i^a \in \mathbb{R}^3$ and gyroscope readings $x_i^\omega \in \mathbb{R}^3$ at time instant $i \in \mathbb{N}$. Given the raw IMU readings sequence $X_i = \{x_i, x_{i+1}, \dots, x_{i+L-1}\} \in \mathbb{R}^{L \times 6}$ where L denotes the observation scope, the test statistics y_i can be calculated through the function $f(\cdot)$. The stationary hypothesis upon an empirical threshold γ can be determined as

$$y_i = f(X_i) \begin{cases} \leq \gamma & \text{for gait is stationary} \\ > \gamma & \text{for gait is mobile.} \end{cases} \quad (11)$$

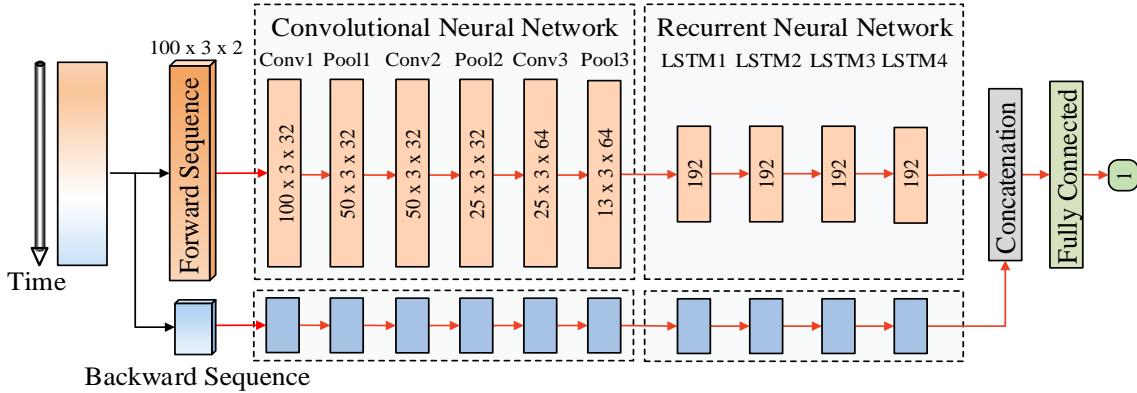


Fig. 3. Architecture of Symmetrical-Net with detailed configuration. Input refers to IMU data surrounding the undetermined point at which the sequence is separated equally to be fed into the corresponding branch. Output refers to the probability of being a stationary point. The identical forward-branch (red) and backward-branch (blue) are formed to represent symmetry.

Algorithm 1: ZUPT-aided INS

```

Input :  $\omega^{b_k}, a^{b_k}, C_{b_{k-1}}^n, v_{k-1}^n, p_{k-1}^n, P_{k-1}$ 
output:  $C_{b_k}^n, v_k^n, p_k^n$ 
begin
     $C_{b_k}^n \leftarrow (1);$ 
     $v_k^n \leftarrow (2);$ 
     $p_k^n \leftarrow (3);$ 
     $\hat{P}_k = F_k P_{k-1} F_k^T + Q;$ 
    if in the stationary phase then
         $K_k = \hat{P}_k H^T (H \hat{P}_k H^T + R)^{-1};$ 
         $\delta x_k = K_k v_k^{nT} = [\delta \phi_k, \delta p_k, \delta v_k]^T;$ 
         $P_k = (I - K_k H) \hat{P}_k;$ 
         $C_{b_k}^n \leftarrow (I + [\delta \phi_k] \times) C_{b_k}^n;$ 
         $p_k^n \leftarrow p_k^n - \delta p_k;$ 
         $v_k^n \leftarrow v_k^n - \delta v_k;$ 
         $\delta x_k \leftarrow \bar{0};$ 
    end
end

```

There are three main ways to define function $f(\cdot)$ in conventional fixed threshold detector.

- Acceleration Moving Variance (AMV) [20], [34]

$$f(X_i) = \frac{1}{\sigma_a^2 L} \sum_{k=i}^{i+L-1} \|x_k^a - \bar{X}_i^a\|^2 \quad (12)$$

where $\sigma_a \in \mathbb{R}^1$ denotes the standard deviation of accelerometer noise, and $\|\cdot\|$ is 2-norm of the vector. Further,

$$\bar{X}_i^a = \frac{1}{L} \sum_{k=i}^{i+L-1} x_k^a \quad (13)$$

indicates the mean acceleration of sequence X_i .

- Acceleration Magnitude (MAG) [19], [35]–[37]

$$f(X_i) = \frac{1}{\sigma_a^2 L} \sum_{k=i}^{i+L-1} (\|x_k^a\| + g^n)^2 \quad (14)$$

- Angular Rate Energy (ARE) [21], [38], [39]

$$f(X_i) = \frac{1}{\sigma_\omega^2 L} \sum_{k=i}^{i+L-1} \|x_k^\omega\|^2 \quad (15)$$

where $\sigma_\omega \in \mathbb{R}^1$ denotes the standard deviation of gyroscope noise.

The major problem for the fixed threshold detector is that the proper threshold varies with the individual and pace, whichever the function $f(\cdot)$ is chosen. To address this problem, we employ the deep learning method as the function $f(\cdot)$, which avoids seeking appropriate threshold manually.

C. Symmetrical-Net as Detector

1) *Inspiration for Symmetrical-Net:* By considering IMU data as a waveform image, the utilization of CNN to extract high-level representation makes sense. Also, pedestrian gait is a time-series process. It is necessary to derive connections among consecutive sampling points, hence the employment of Recurrent Neural Networks (RNN). The proposed Symmetrical-Net is, essentially, a deep RCNNs that cascades two different neural networks.

Although literature [22], [25], [26], [40] presented zero velocity detection utilizing learning-based approaches, they only focus on raw readings before the time instant. As shown in Fig. 3, a symmetrical structure is proposed to extend the observing scope of neural network. Given a $2l-1$ long sequence $X_i = \{x_{i-l+1}, \dots, x_{i-1}, x_i, x_{i+1}, \dots, x_{i+l-1}\}$, Symmetrical-Net serving as the function $f(\cdot)$, calculates the probability of time instance i being stationary. The probability y_i is in the range of 0 to 1. Upon a constant threshold 0.5, the stationary phase can be determined as

$$y_i = f(X_i) \begin{cases} \leq 0.5 & \text{for gait is stationary} \\ > 0.5 & \text{for gait is mobile.} \end{cases} \quad (16)$$

The whole Symmetrical-Net contains two identical branches, namely forward-branch and backward-branch. It is noteworthy that X_i has to be arranged in the order $X_i^{forward} = \{x_{i-l+1}, \dots, x_{i-1}, x_i\}$ and $X_i^{backward} = \{x_{i+l-1}, \dots, x_{i+1}, x_i\}$ to input corresponding branch. It should

be emphasized that our proposed symmetrical framework is different from the bidirectional recurrent neural network (BRNN) [41] and the Siamese Network [42] since 1) inputs of forward sequence and backward sequence of BRNN, indeed, are the same identical in the inverse direction, while the inputs of forward branch and backward branch in the Symmetrical-Net are entirely different; and 2) the Siamese Network learns a similarity metric from data, which can be used to compare or match two samples, while two samples are mapped to the low-dimensional space by the neural networks with shared parameter. It means that there is, indeed, only one neural network in the Siamese Network to learn the representations instead of two parallel branches in the Symmetrical-Net.

2) CNN Based Feature Extraction: In this paper, window size l is set to 100, considering the average duration of a stride is 0.5s, and our IMU sample rate is 200Hz. Thus, the observation is in an approximate range of an entire stride. Note that too long or narrow interval of input sequences will affect the neural network performance. In addition, the IMU measurement suffers from the additive white Gaussian noise. The bias of the sensor, namely the random walk, is the integral of the white noise in different noise density with respect to time. To enable Symmetrical-Net to learn this IMU noise model, the input sequence X_i is the raw IMU readings, which makes the model more robust [43].

Input sequences $X_i^{forward}$ and $X_i^{backward}$ are reshaped into the size of 100 (window size) \times 3 (X, Y and Z axis data) \times 2 (gyroscope and accelerometer), as the input tensor for each branch of CNN. It has three convolutional layers extracting features from input data, and each layer is followed by a Max-Pooling layer reducing the feature size and enhancing the robustness of the model. The size of receptive fields is all set to 15×3 . The utilization of the same-padding is to preserve the spatial dimension of the tensor after each convolution. The increasing number of channels (i.e., the number of filters for feature detection) is set to learn various features.

3) LSTM Based Contextual Modelling: The outcome of CNN is time-series data because the operation of filtering is in sequence. The following RNN is designed to conduct sequential learning. Note that output of CNN is $y_{cnn} \in \mathbb{R}^{13\times 3\times 64}$ that has to be reshaped into $X_{rnn} \in \mathbb{R}^{13\times 192}$ so as to input RNN for both branches.

Although standard RNN allows information to persist, the performance worsens as the sequence length on the increase is due to the gradient vanishing problem. LSTM, a special kind of RNN, is an efficient long-term dependencies model. The kernel of LSTM is the cell state controlled by three gates (i.e., forget gate, input gate, and output gate). As shown in Fig. 4, with the input S_t , the hidden state h_{t-1} , and cell state c_{t-1} , variables update at t time step on the basis of:

$$f_t = \sigma(w_{fx}S_t + w_{fh}h_{t-1} + b_f) \quad (17)$$

$$i_t = \sigma(w_{ix}S_t + w_{ih}h_{t-1} + b_i) \quad (18)$$

$$g_t = \tanh(w_{gx}S_t + w_{gh}h_{t-1} + b_g) \quad (19)$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \quad (20)$$

$$o_t = \sigma(w_{ox}S_t + w_{oh}h_{t-1} + b_o) \quad (21)$$

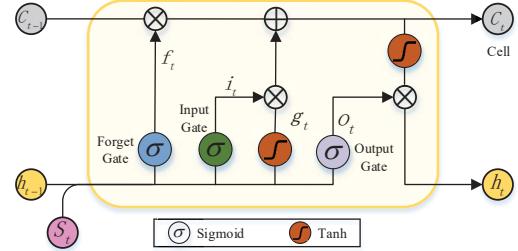


Fig. 4. Internal structure of LSTM contains three gates and the cell unit. The forward propagation at time t , meanwhile, presents in this diagram.

$$h_t = o_t \odot \tanh(c_t) \quad (22)$$

where σ and \tanh represent the activation in sigmoid non-linearity and hyperbolic tangent non-linearity. \odot is dot product operation on two vectors. w and b denote corresponding weight matrixes and bias vectors, which are parameters to be trained. (17), (18) and (21) give the calculation principles of forget gate f_t , input gate i_t and output gate o_t , respectively. (19) creates the candidate values g_t meaning the information that could be stored in the cell state. The cell state c_t updates according to (20) that determines the state of hidden unit h_t (i.e., the output at t time) by (22).

As shown in Fig. 3, the Symmetrical-Net contains four LSTM layers with 192 units per layer. Each RNN block's output is the hidden unit state of the last layer at the last time step. The aforementioned order of $X_i^{forward}$ and $X_i^{backward}$ guarantees that output $y_{rnn}^{forward}, y_{rnn}^{backward} \in \mathbb{R}^{192}$ are all the features with respect to the undetermined point i . Two vectors are concatenated to be fed into a single fully-connected layer that outputs the probability y_i .

4) Training: Our model is implemented in Keras, and the training is conducted on an NVIDIA Geforce Titan XP GPU. The batch size is set to 512, and we employ Adam [44] with the learning rate $\times 10^{-4}$ (with reduction by half every 10 epochs) to train the neural network for up to 200 epochs (with early stopping). For each raw IMU sequence X_i , the corresponding gait condition is $Y_i \in \{0, 1\}$ that also refers to the label of the input sequence. The optimal parameters θ^* of the Symmetrical-Net is:

$$\theta^* = \arg \max_{\theta} p(Y_i | X_i; \theta). \quad (23)$$

Our model learns parameters θ by the process of minimizing the Mean Square Error (MSE) between estimated probability y_i and label Y_i :

$$\theta^* = \arg \min_{\theta} \frac{1}{N} \sum_{i=1}^N \|y_i - Y_i\|^2 \quad (24)$$

where N is the number of training samples.

As mentioned before, although the stance phase occupies 60% of the whole stride, there is only 40% of the stance phase in which the foot is entirely stationary. Thus, the negative samples (i.e., non-zero velocity points) are almost three times as many as positive samples (i.e., zero velocity points) in each trail. We define λ is the ratio of negative samples to the positive

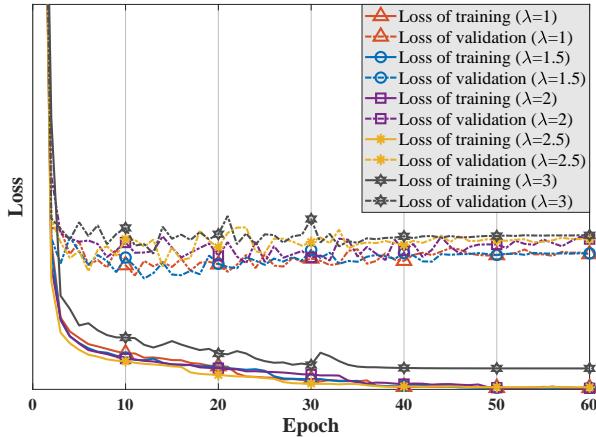


Fig. 5. The losses of training and validation on different ratios of negative samples to positive samples.

samples, and the influence of λ on training the neural network is discussed. Fig. 5 shows the training and validation losses under the ratio λ from 1 to 3. Losses on both training and validation are highest when data is not balanced ($\lambda = 3$). With the decrease in λ , the downward trend in converged losses indicates that the training procedure on Symmetrical-Net becomes better. While the training losses are almost the same in terms of λ ranging between 1 and 2.5 after 40 epochs, it is more appropriate to choose λ from 1 or 1.5 to balance the negative and positive samples in training data. This owes much to the lower validation losses when λ is 1 or 1.5, which indicates the alleviation of overfitting. In this paper, the ratio is 1.5, to ensure the sufficiency of training data.

IV. EVALUATION AND ANALYSIS

A. Data Preparation

A Next Generation IMU (NGIMU) is used as our equipment for measurement. It is attached to the right foot of pedestrian as shown in Fig. 6. The build-in 3-axis accelerometer and gyroscope are at the sampling rate of 200Hz. We attribute the abandonment of magnetometer to the distortion of the magnetic field when testing indoors. The detailed specifications of NGIMU are shown in Table I.



Fig. 6. Foot-mounted NGIMU

Extensive data containing three types of motion is collected among 27 individuals in 7 different sites. Note that the entire 27 individuals separate into 20 for training and 7 for testing,

TABLE I
SPECIFICATIONS OF NGIMU

	Accelerometer	Gyroscope
Axes	3	3
Sample Rate	Up to 400 Hz	Up to 400 Hz
Acquisition Range	$\pm 16g$	$\pm 2000^\circ/s$
Communication Interfaces	USB Wi-Fi: 802.11n; 5 GHz; AP or client mode SD card	
Power Management	Power from USB or battery Battery: 1000 mAh; charging via USB Battery measurement: Contain percentage, time to empty, voltage, current; sample rate at 5 Hz	

as shown in Table II, which guarantees the evaluation of the model robustness.

TABLE II
DISTRIBUTION OF COLLECTED DATA

Usage	Motion	Total Length (m)	Individual Amount
Training	Walking	6130	
	Fast Walking	4107	20
	Running	1920	
Testing	Walking	2150	
	Fast Walking	1560	7
	Running	1070	

Erroneous labels will cause an adverse effect on the neural network performance. Considering that gyroscope readings hold the most reliable information for zero velocity detection [10], our strategy for labeling zero velocity intervals follows the ARE in (15):

$$y_i = \sqrt{(\omega_x^i)^2 + (\omega_y^i)^2 + (\omega_z^i)^2} \begin{cases} \leq \gamma & \text{label is 1 (stationary)} \\ > \gamma & \text{label is 0 (moving)} \end{cases} \quad (25)$$

where ω_x^i , ω_y^i , ω_z^i denote the 3-axis angular velocity at time i measured by gyroscope. When we gathering the trails, participants moved along the routes as planned. Recording the marker attached to the floor, the ground truth for each trail is well measured. In each trajectory, pedestrian return to the start point, which gives efficient canon to assess the labels. Thus, undistorted trails that satisfy the smallest distance between the start point and the endpoint reflect eligible labeling. Fig. 7 is the process of labeling a certain trail. Trajectory tends to be distorted because the excessively low (e.g., $\gamma = 0.35$) or high (e.g., $\gamma = 2$) threshold results in false zero velocity points pick. After confirming the proper threshold (e.g., $\gamma = 0.7$), some zero velocity points with a short continuous length need to be filtered, which further refines the labels.

B. Experimental Results

Our proposed adaptive detector makes a very positive contribution to conserving the manual labour compared with the fixed threshold method. The accuracy and robustness of Symmetrical-Net can be evaluated from two aspects: the

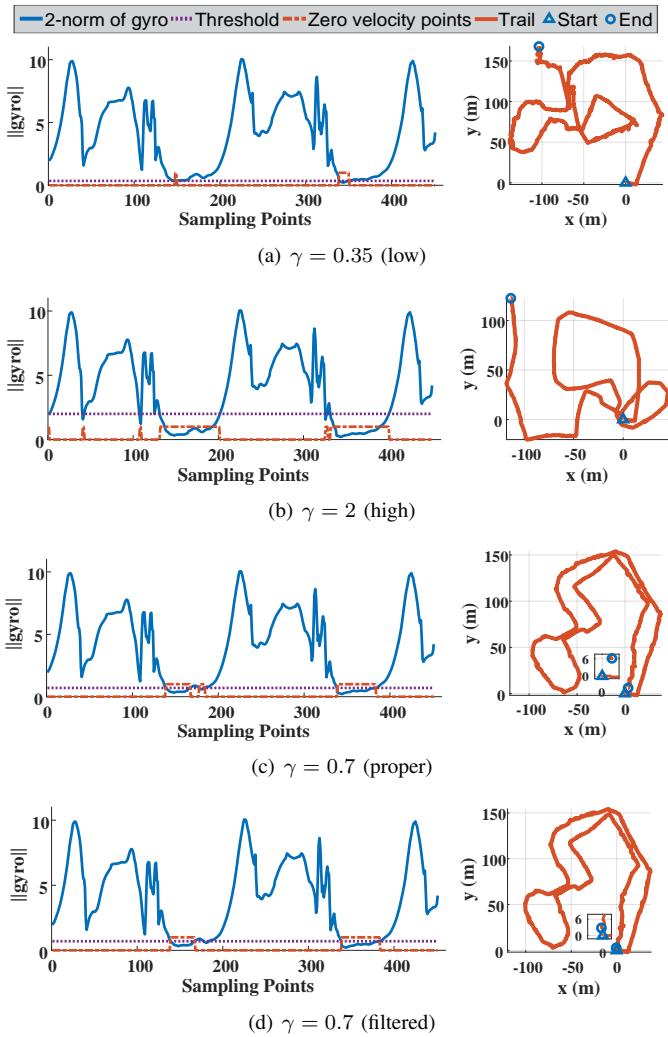


Fig. 7. From the top to bottom shows the process of tuning the threshold γ to label the zero velocity points: $\gamma = 0.35$ (low), $\gamma = 2$ (high), $\gamma = 0.7$ (proper), $\gamma = 0.7$ (filtered).

classification accuracy and the performance of assisting INS. Experiments are conducted to verify two characteristics of the Symmetrical-Net: the utilization of contextual information and the extension of observation scope. Five variants of the Symmetrical-Net shown in Table III are compared to prove the best hyperparameter setting. In the Table III, Cx denotes a convolutional layer with x kernels, and Lx denotes a LSTM layer contains x hidden states. Among them, Symmetrical-Net5 is the reference architecture and the number of either CNN layers or LSTM layers is adjusted in the other four competing models. A CNN-only model (i.e., AZUPT in [40]) and an asymmetric RCNNs model (i.e., the forward branch of the Symmetrical-Net5) are also used for comparison. Note that all the neural networks share the same training set.

1) *Classification Accuracy*: Table IV gives the classification accuracy of seven adaptive detectors on the test data. Assessment criteria are precision, recall, and F1-score, which are commonly used in the classification task [24]. They are defined

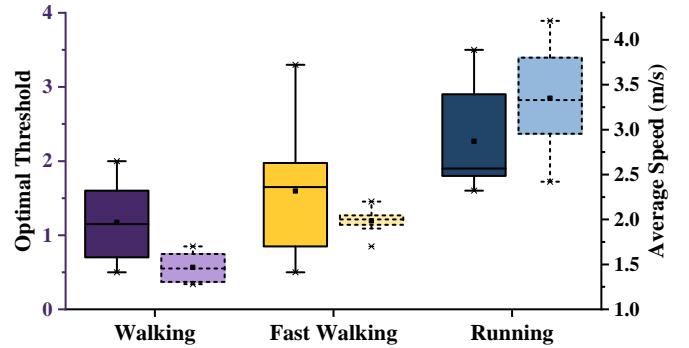


Fig. 8. Statistics about the optimal thresholds (solid line) and average speed (dashed line) for three motion patterns based on labeling all 87 collected trails.

TABLE III
ARCHITECTURES OF DIFFERENT NEURAL NETWORKS

Model	Architecture	Symmetrical Framework
Symmetrical-Net1	$C32-C32-C64-L192-L192-L192$	YES
Symmetrical-Net2	$C32-C32-C64-L192-L192-L192-L192$	YES
Symmetrical-Net3	$C32-C64-L192-L192-L192-L192$	YES
Symmetrical-Net4	$C16-C32-C32-C64-L192-L192-L192-L192$	YES
Symmetrical-Net5	$C32-C32-C64-L192-L192-L192-L192$	YES
RCNNs	$C32-C32-C64-L192-L192-L192-L192$	NO
AZUPT	$C32-C64$	NO

as follows:

$$Precision = \frac{TP}{\sum Positive\ Prediction} = \frac{TP}{TP + FP} \quad (26)$$

$$Recall = \frac{TP}{\sum Positive\ Sample} = \frac{TP}{TP + FN} \quad (27)$$

$$F_1 = \frac{Precision \cdot Recall}{Precision + Recall} \quad (28)$$

where TP , FP , and FN denote the true-positive, false-positive, and false-negative error, respectively. In terms of zero velocity detection, FP detection leads to input wrong measurements to ESKF, which causes a more severe impact than FN detection. Therefore, precision plays a vital role among these criteria.

By considering additional contextual information, the utilization of RNN improves detection accuracy. It is confirmed by the fact that the classification accuracy of RCNNs exceeds the AZUPT, especially in fast walking. Results also show that in three motion patterns, Symmetrical-Nets with five different configurations outperform the competing architectures (i.e., CNN-only and asymmetric RCNNs) under all three assessment criteria. Not only does the Symmetrical-Net consider the contextual information, but our proposed method also extends the observation range.

By comparing the classification accuracy of Symmetrical-Net1 and the reference model, namely Symmetrical-Net5, while the reduction in the number of LSTM layers slightly

TABLE IV
THE CLASSIFICATION ACCURACY

Model	Walking				Fast Walking				Running		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score		
Symmetrical-Net1	0.930	0.9075	0.910	0.930	0.9125	0.915	0.895	0.877	0.861		
Symmetrical-Net2	0.907	0.891	0.893	0.893	0.883	0.885	0.902	0.897	0.894		
Symmetrical-Net3	0.931	0.917	0.920	0.923	0.913	0.913	0.918	0.912	0.905		
Symmetrical-Net4	0.928	0.905	0.910	0.926	0.913	0.917	0.910	0.894	0.884		
Symmetrical-Net5	0.928	0.926	0.927	0.920	0.918	0.919	0.923	0.920	0.921		
RCNNs	0.910	0.875	0.880	0.908	0.898	0.900	0.905	0.905	0.905		
AZUPT	0.910	0.855	0.865	0.873	0.883	0.875	0.900	0.895	0.900		

elevates the performance on the walking and fast walking, it causes the results under running fall sharply. Nevertheless, if we stack more LSTM layers, the results of Symmetrical-Net2 illustrate that errors in detecting zero velocity intervals under running are not significantly alleviated. Still, the accuracy under the other two motions is severely affected. In terms of the configuration of the CNN, the results of Symmetrical-Net3, Symmetrical-Net4 are less distinct from Symmetrical-Net5 compared with the variation attributed to the change to the LSTM settings. In other words, deep LSTM layers are more prone to overfitting. Through a trade-off between the model performance under all motion patterns, we adopt the Symmetrical-Net5, and the Symmetrical-Net in the following experiments refers to the neural network deployed as Symmetrical-Net5.

2) *Performance of Assisting INS*: We also evaluate the performance of INS assisted by different detectors. Under the guideline of statistics shown in Fig. 8, the empirical thresholds for the fixed threshold detector are the average of optimal threshold in each motion: $\gamma = 1.2$ for walking, $\gamma = 1.7$ for fast walking, and $\gamma = 2.2$ for running.

The ground truth in this paper is generated from the markers on the ground that guide the routes of the participants according to plan. It means that the ground truth only contains the position of pedestrians. Although the rotational error evaluation is of the importance to navigation task, only the translation (i.e., the position change between the consecutive IMU readings) is considered in the following experiments. To obtain the precise rotation of reference, Iterative Closest Point (ICP) based on the accurate point clouds provided by a 3D laser scanner is preferred [45].

We calculate the positioning error of reconstructed trajectories, and plot the cumulative distribution functions (CDFs) in Fig. 9. The proposed detector outperforms other competing approaches in all three motions, and the error of RCNNs is lower than AZUPT in terms of high-pace movement. It demonstrates that the time-series constraint improves the robustness. For INS assisted by the fixed threshold detector, accurate pose estimation in fast walking is most challenging. This is attributed to the blurred differences between fast walking and the other two motions, which makes the threshold hard to decide. Whereas the learning-based algorithms analyzing features in high-dimension are more efficient and adaptive.

The accumulated error causes the trails to diverge with the movement. Therefore, evaluating the error against the path length is essential for positioning systems that estimate the relative displacement. We calculate the translation error

following:

$$e_L = \frac{\sum_{i=1}^N \sqrt{(\Delta x'_i - \Delta x_i)^2 + (\Delta y'_i - \Delta y_i)^2}}{L} \times 100\% \quad (29)$$

where $\Delta x'$ and $\Delta y'$ are translation during a sampling period in the 2D plane estimated by INS. Δx and Δy are the reference translation. Within the traveled length L , there are N samples. The e_L is capable of describing the estimated translation error with regard to the moving process. To represent the translation error of INS assisted by four different detectors in three motions, Fig. 10 depicts the average translation error on the length of 50m to 450m. The Symmetrical-Net, as the zero velocity detector, is consistently better than the other three detectors. With the length of trajectory increasing, the translation error of the proposed approach is in a downtrend. Compared with error CDFs, e_L more obviously reveals the improvement in pose estimation using adaptive detectors when the pedestrian walking. This is because the translation accuracy of the fixed threshold is competitive only at the first 100m of a trail.

Several trails of INS estimating the test data are plotted in Fig. 11 that qualitatively demonstrates the aforementioned results. The reconstructed trajectories of ZUPT-aided INS with the fixed threshold detector suffer from considerable divergent, especially for the fast walking and running, since 1) the thresholds used in the experiments are selected according to statistics, while the threshold shows high dynamic range when pedestrian run and fast walk; and 2) the ZUPT-aided INS used in the experiments is a quite general algorithm, which lacks the knowledge of the ambient magnetic field and the assistance of extra compensating methods (e.g., zero angular rate update [46] and heuristic drift reduction [47]). Nevertheless, we take such basic ZUPT-aided INS with the fixed threshold as the baseline to evaluate our algorithm proposed to detect the stationary phase of a gait. It is reasonable to infer that the proposed method still works in other systems involving pedestrian gait phase segmentation. While the AZUPT and RCNNs model outperform the fixed threshold, which is still not as good as our approach, the divergent of two adaptive models arises during running, as shown in Fig. 11(f). Without enough training samples covering high-pace motion, the model suffers from high drifts. This problem can be addressed by Symmetrical-Net. It owes much to the addition of the backward sequence, which can be regarded as a kind of data augmentation. Overall, RNN that considers the time-series representation improves the zero velocity detection accuracy. The devised symmetrical framework takes sufficient information around each undetermined point, enhancing the

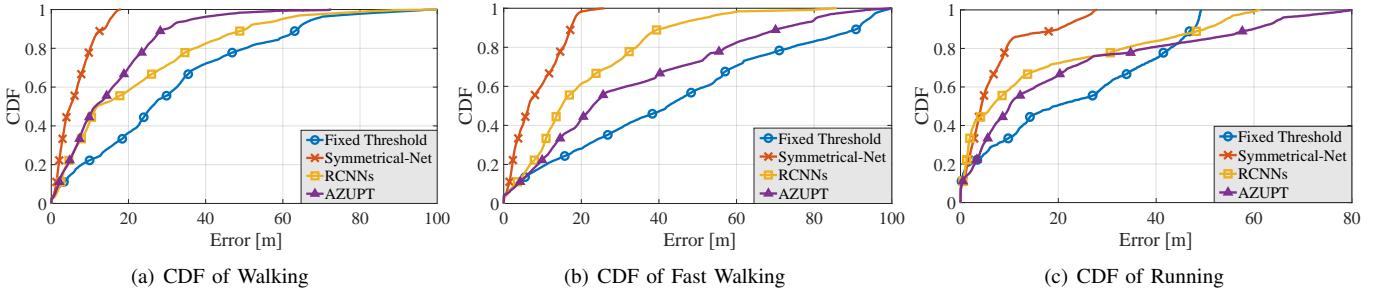


Fig. 9. Error CDFs of three motions representing the performance of four different detector on the test data.

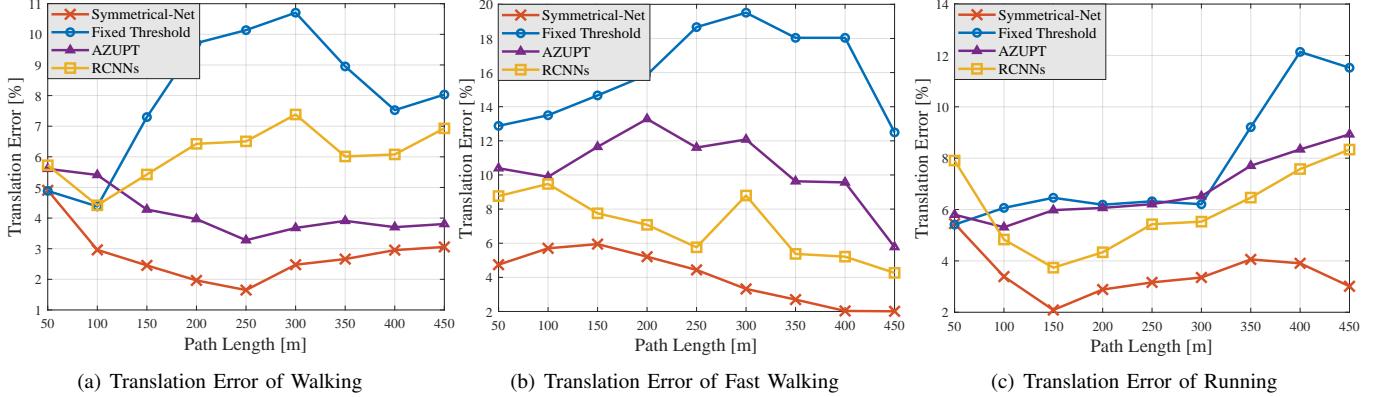


Fig. 10. The translation error against the trail length in three motions.

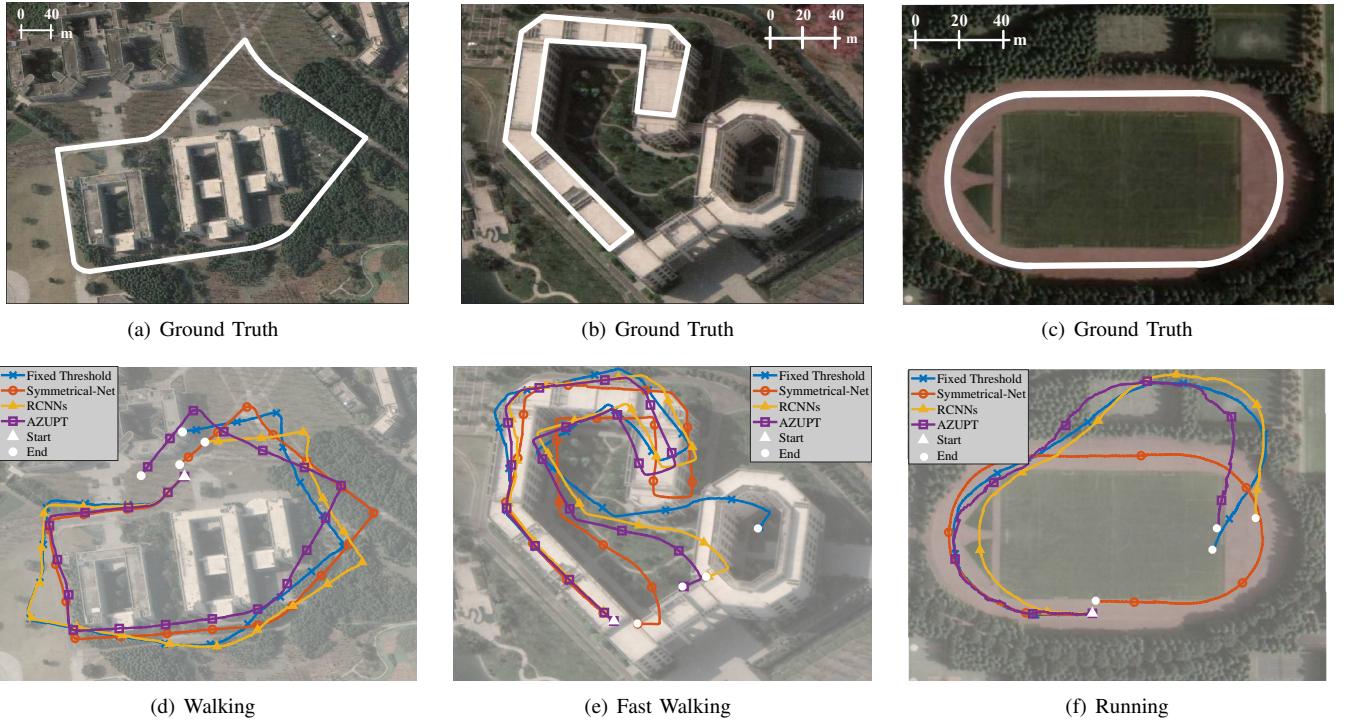


Fig. 11. Results of ZUPT-aided INS in three motions showing ground truth (top), reconstructed trails (bottom). We compare the proposed Symmetrical-Net with other three different detectors including two adaptive detectors.

model robustness, even on the small training data.

3) Computational Efficiency: In terms of navigation tasks, it is crucial for the model to satisfy real-time positioning in

practical applications. We assess time complexity, and scale of parameters by comparing the Symmetrical-Net with four neural networks [48]–[51] that are capable of applying on

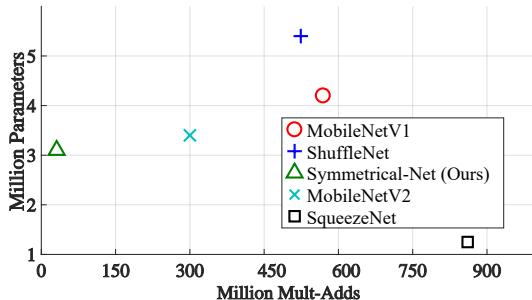


Fig. 12. Comparison of time complexity and parameter count in Symmetrical-Net and several state-of-the-art models for mobile applications.

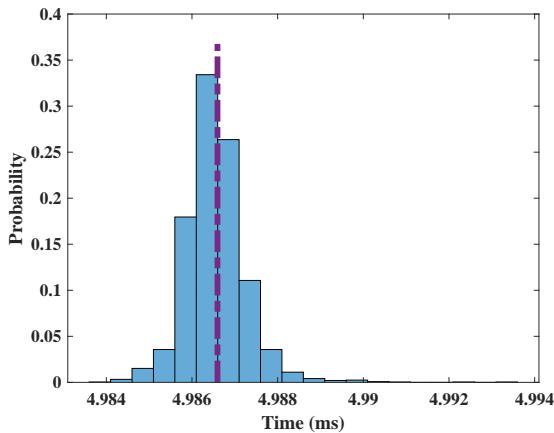


Fig. 13. The time consumption frequency histogram of Symmetrical-Net on CPU. The average time consumption is 4.9866 ms (purple dash line), calculated from 2500 input sequences.

the mobile platforms. As shown in Fig. 12, Symmetrical-Net is lighter than competing approaches. Also, we record the online run time of 2500 input samples conducted on Intel(R) Core(TM) i7-8700 CPU@3.20GHz, and the statistic is shown in Fig. 13. The average time consumption is 4.9866 ms per input sequence, which is lower than the sampling period at 200Hz. With parallel computing, offline processing is much more efficient.

V. CONCLUSION

In this paper, we demonstrate the merits of a novel zero velocity detection technique based on the application of deep learning. Leveraging the power of RCNNs, we design a model combining CNN and LSTM, which can adaptively detect the zero velocity. The symmetrical structure is conducted to extend the observation scope, which further improves the robustness and classification accuracy. Besides, the reconstructed trajectories based on ZUPT-aided PDR with the assistant of proposed Symmetrical-Net as a zero velocity detector also show significant improvement.

REFERENCES

- [1] A. Sciarrone, C. Fiandrino, I. Bisio, F. Lavagetto, D. Kliazovich, and P. Bouvry, “Smart probabilistic fingerprinting for indoor localization over fog computing platforms,” in *2016 5th IEEE International Conference on Cloud Networking (Cloudnet)*, pp. 39–44, IEEE, 2016.
- [2] L. Pei, R. Chen, J. Liu, T. Tenhunen, H. Kuusniemi, and Y. Chen, “An inquiry-based bluetooth indoor positioning approach for the finnish pavilion at shanghai world expo 2010,” in *IEEE/ION Position, Location and Navigation Symposium*, pp. 1002–1009, IEEE, 2010.
- [3] F. Lemic, A. Behboodi, V. Handziski, and A. Wolisz, “Experimental decomposition of the performance of fingerprinting-based localization algorithms,” in *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 355–364, IEEE, 2014.
- [4] R. Zhang, F. Hoflinger, and L. Reindl, “Inertial sensor based indoor localization and monitoring system for emergency responders,” *IEEE Sensors Journal*, vol. 13, no. 2, pp. 838–848, 2013.
- [5] J. Wahlström, P. Porto Buarque de Gusmão, A. Markham, and N. Trigoni, “Map-aided navigation for emergency searches,” in *2019 15th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, pp. 25–32, 2019.
- [6] J. Rantakokko, J. Rydell, P. Strömbeck, P. Händel, J. Callmer, D. Törnqvist, F. Gustafsson, M. Jobs, and M. Grudn, “Accurate and reliable soldier and first responder indoor positioning: multisensor systems and cooperative localization,” *IEEE Wireless Communications*, vol. 18, no. 2, pp. 10–18, 2011.
- [7] E. Foxlin, “Pedestrian tracking with shoe-mounted inertial sensors,” *IEEE Computer graphics and applications*, no. 6, pp. 38–46, 2005.
- [8] N. Bai, Y. Tian, Y. Liu, Z. Yuan, Z. Xiao, and J. Zhou, “A high-precision and low-cost imu-based indoor pedestrian positioning technique,” *IEEE Sensors Journal*, vol. 20, no. 12, pp. 6716–6726, 2020.
- [9] Ö. Bebek, M. A. Suster, S. Rajgopal, M. J. Fu, X. Huang, M. C. Çavuşoğlu, D. J. Young, M. Mehregany, A. J. van den Bogert, and C. H. Mastrangelo, “Personal navigation via high-resolution gait-corrected inertial measurement units,” *IEEE Transactions on Instrumentation and Measurement*, vol. 59, no. 11, pp. 3018–3027, 2010.
- [10] I. Skog, J.-O. Nilsson, and P. Händel, “Evaluation of zero-velocity detectors for foot-mounted inertial navigation systems,” in *2010 International Conference on Indoor Positioning and Indoor Navigation*, pp. 1–6, IEEE, 2010.
- [11] X. Niu, Y. Li, J. Kuang, and P. Zhang, “Data fusion of dual foot-mounted imu for pedestrian navigation,” *IEEE Sensors Journal*, vol. 19, no. 12, pp. 4577–4584, 2019.
- [12] J. Wahlström and I. Skog, “Fifteen years of progress at zero velocity: A review,” *IEEE Sensors Journal*, pp. 1–1, 2020.
- [13] J. Perry, J. R. Davids, et al., “Gait analysis: normal and pathological function,” *Journal of Pediatric Orthopaedics*, vol. 12, no. 6, p. 815, 1992.
- [14] M. W. Whittle, *Gait analysis: an introduction*. Butterworth-Heinemann, 2014.
- [15] X. Liu, S. Wang, T. Zhang, R. Huang, and Q. Wang, “A zero-velocity detection method with transformation on generalized likelihood ratio statistical curve,” *Measurement*, vol. 127, pp. 463–471, 2018.
- [16] Y. Wang, A. Chernyshoff, and A. M. Shkel, “Error analysis of zupt-aided pedestrian inertial navigation,” in *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 206–212, IEEE, 2018.
- [17] Q. Wang, M. Cheng, A. Noureddin, and Z. Guo, “Research on the improved method for dual foot-mounted inertial/magnetometer pedestrian positioning based on adaptive inequality constraints kalman filter algorithm,” *Measurement*, vol. 135, pp. 189–198, 2019.
- [18] I. Skog, P. Handel, J.-O. Nilsson, and J. Rantakokko, “Zero-velocity detection and algorithm evaluation,” *IEEE transactions on biomedical engineering*, vol. 57, no. 11, pp. 2657–2666, 2010.
- [19] A. M. Sabatini, C. Martelloni, S. Scapellato, and F. Cavallo, “Assessment of walking features from foot inertial sensing,” *IEEE Transactions on biomedical engineering*, vol. 52, no. 3, pp. 486–494, 2005.
- [20] P. H. Veltink, H. J. Bussmann, W. De Vries, W. J. Martens, R. C. Van Lummen, et al., “Detection of static and dynamic activities using uniaxial accelerometers,” *IEEE Transactions on Rehabilitation Engineering*, vol. 4, no. 4, pp. 375–385, 1996.
- [21] J. M. Jasiewicz, J. H. Allum, J. W. Middleton, A. Barriskill, P. Condie, B. Purcell, and R. C. T. Li, “Gait event detection using linear accelerometers or angular velocity transducers in able-bodied and spinal-cord injured individuals,” *Gait & posture*, vol. 24, no. 4, pp. 502–509, 2006.
- [22] B. Wagstaff, V. Peretroukhin, and J. Kelly, “Improving foot-mounted inertial navigation through real-time motion classification,” in *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–8, IEEE, 2017.
- [23] S. Y. Park, H. Ju, and C. G. Park, “Stance phase detection of multiple actions for military drill using foot-mounted imu,” *sensors*, vol. 14, p. 16, 2016.

- [24] R. Zhu, Z. Xiao, Y. Li, M. Yang, Y. Tan, L. Zhou, S. Lin, and H. Wen, "Efficient human activity recognition solving the confusing activities via deep ensemble learning," *IEEE Access*, vol. 7, pp. 75490–75499, 2019.
- [25] B. Wagstaff and J. Kelly, "Lstm-based zero-velocity detection for robust inertial navigation," in *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–8, IEEE, 2018.
- [26] B. Wagstaff, V. Peretroukhin, and J. Kelly, "Robust data-driven zero-velocity detection for foot-mounted inertial navigation," *IEEE Sensors Journal*, vol. 20, no. 2, pp. 957–967, 2019.
- [27] M. Edel and E. Köppé, "An advanced method for pedestrian dead reckoning using blstm-rnns," in *2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–6, IEEE, 2015.
- [28] T. D. Barfoot, *State estimation for robotics*. Cambridge University Press, 2017.
- [29] J.-O. Nilsson, I. Skog, and P. Händel, "A note on the limitations of zupts and the implications on sensor error modeling," in *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 13–15th November 2012, 2012.
- [30] D. S. Colomar, J.-O. Nilsson, and P. Händel, "Smoothing for zupt-aided inss," in *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pp. 1–5, IEEE, 2012.
- [31] S. Thrun, "Probabilistic robotics," *Communications of the ACM*, vol. 45, no. 3, pp. 52–57, 2002.
- [32] S. M. Kay, *Fundamentals of statistical signal processing*. Prentice Hall signal processing series, Upper Saddle River, NJ: Prentice Hall PTR, 1993.
- [33] J. Wahlström, A. Markham, and N. Trigoni, "Footslam meets adaptive thresholding," *IEEE Sensors Journal*, vol. 20, no. 16, pp. 9351–9358, 2020.
- [34] S. Kwakkel, G. Lachapelle, and M. Cannon, "Gnss aided in situ human lower limb kinematics during running," in *Proc. of ION GNSS*, 2008.
- [35] S. Godha and G. Lachapelle, "Foot mounted inertial system for pedestrian navigation," *Measurement Science and Technology*, vol. 19, no. 7, p. 075202, 2008.
- [36] S. Godha, G. Lachapelle, and M. E. Cannon, "Integrated gps/ins system for pedestrian navigation in a signal degraded environment," in *Ion gnss*, vol. 2006, pp. 26–29, 2006.
- [37] B. Krach and P. Robertson, "Integration of foot-mounted inertial sensors into a bayesian location estimation framework," in *2008 5th Workshop on Positioning, Navigation and Communication*, pp. 55–61, IEEE, 2008.
- [38] R. Feliz Alonso, E. Zalama Casanova, and J. Gómez García-Bermejo, "Pedestrian tracking using inertial sensors," 2009.
- [39] L. Ojeda and J. Borenstein, "Non-gps navigation for security personnel and first responders," *Journal of Navigation*, vol. 60, no. 3, p. 391, 2007.
- [40] X. Yu, B. Liu, X. Lan, Z. Xiao, S. Lin, B. Yan, and L. Zhou, "Azupt: Adaptive zero velocity update based on neural networks for pedestrian tracking," in *2019 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–6, 2019.
- [41] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.
- [42] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, pp. 539–546, IEEE, 2005.
- [43] M. A. Esfahani, H. Wang, K. Wu, and S. Yuan, "Aboldeepio: A novel deep inertial odometry network for autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 5, pp. 1941–1950, 2019.
- [44] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [45] M. Ramezani, Y. Wang, M. Camurri, D. Wisth, M. Mattamala, and M. Fallon, "The newer college dataset: Handheld lidar, inertial and vision with ground truth," *arXiv preprint arXiv:2003.05691*, 2020.
- [46] A. R. Jiménez, F. Seco, J. C. Prieto, and J. Guevara, "Indoor pedestrian navigation using an ins/ekf framework for yaw drift reduction and a foot-mounted imu," in *2010 7th Workshop on Positioning, Navigation and Communication*, pp. 135–143, IEEE, 2010.
- [47] J. Borenstein, L. Ojeda, and S. Kwanmuang, "Heuristic reduction of gyro drift in imu-based personnel tracking systems," in *Optics and Photonics in Global Homeland Security V and Biometric Technology for Human Identification VI*, vol. 7306, p. 73061H, International Society for Optics and Photonics, 2009.
- [48] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: Alexnet-level accuracy with 50x fewer parameters and; 0.5 mb model size," *arXiv preprint arXiv:1602.07360*, 2016.
- [49] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: An extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6848–6856, 2018.
- [50] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.
- [51] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018.



Mingkun Yang received the B.Eng. Degree from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2018. He is working toward a master's degree in the School of Information and Communication Engineering, UESTC. His research interests focus on the application of machine learning techniques in sensor networks and indoor localization.



Ran Zhu received the B.Eng. Degree from Jilin University, Changchun, China, in 2018. She is currently pursuing a master's degree in the School of Information and Communication Engineering at the University of Electronic Science and Technology of China, Chengdu, China. Her research interests include sensor fusion and intelligent navigation for autonomous robots.



Zhuoling Xiao is an Associate Professor at the University of Electronic Science and Technology of China. He obtained his Ph.D. at the University of Oxford, became a postdoctoral researcher at the University of Oxford. His interests lie in localization protocols for networked sensor nodes and machine learning techniques for sensor networks and localization. Zhuoling has several international patent applications and over 30 papers published in leading journals and conferences, including several best paper awards from leading conferences, including IPSN and EWSN.



Bo Yan received the master's and Ph.D. degrees from the University of Electronic Science and Technology of China (UESTC), Chengdu, China. And now, she is a professor in the School of Information and Communication Engineering at UESTC. Her current research interests lie in embedded system technology, FPGA/ASIC design, and AI for the Internet of Things(AIoT).