IRT-SD-SLE: An Improved Real-time Step Detection and Step Length Estimation using Smartphone Accelerometer

PAMPA SADHUKHAN ¹, Saptadipa Mazumder ¹, Sara Paiva ¹, Chandreyee Chowdhury ¹, Pradip Das ¹, Keshav Dahal ¹, and Xinheng Wang ¹

¹Affiliation not available

October 31, 2023

Abstract

Smartphone sensor-based pedestrian dead reckoning (PDR) systems provide a viable solution to the problem of localization in an infrastructure-less area. Step detection (SD) and step length estimation (SLE), being two fundamental operations of the PDR based localization technique, have drawn many researchers’ attention in the recent time. Most of the existing SD and SLE methods proposed over the years, however, provide either server- or cloud-based solution that consume additional network bandwidth and suffer from increased transmission delay. Moreover, nonavailability of the inertial sensors like gyroscope, magnetometers etc. at every smartphone makes majority of the existing SLE methods less applicable to such devices. To address the above-said issues, in this paper we focus on devising an improved SLE method that would detect the pedestrian’s steps and subsequently estimate the step length in real-time by processing the accelerometer data at the device itself. Our proposed method transforms the measured acceleration values along the earth coordinate system and also applies sliding window meaning to mitigate the negative effects of the smartphone’s orientation and gravitational bias on the accuracy of SD and SLE. The performances of our proposed method are evaluated in terms of accuracy for five different users by taking the device in two different postures (handheld and trouser pocket) under two different walking modes (normal and fast) to demonstrate its efficacy. Moreover, our proposed method also obtains more than 88% accuracy (median) for all participants when the device is placed in trouser pocket under two different walking modes.
IRT-SD-SLE: An Improved Real-time Step Detection and Step Length Estimation using Smartphone Accelerometer

Pampa Sadhukhan, Member, IEEE, Saptadipa Mazumder, Sara Paiva, Senior Member, IEEE, Chandreyee Chowdhury, Member, IEEE, Pradip K. Das, Senior Member, IEEE, Keshav Dahal, Senior Member, IEEE, and Xinheng Wang, Senior Member, IEEE

Abstract—Smartphone sensor-based pedestrian dead reckoning (PDR) systems provide a viable solution to the problem of localization in an infrastructure-less area. Step detection (SD) and step length estimation (SLE), being two fundamental operations of the PDR based localization technique, have drawn many researchers' attention in the recent time. Most of the existing SD and SLE methods proposed over the years, however, provide either server- or cloud-based solution that consume additional network bandwidth and suffer from increased transmission delay. Moreover, non-availability of the inertial sensors like gyroscope, magnetometers etc. at every smartphone makes majority of the existing SLE methods less applicable to such devices. To address the above-said issues, in this paper we focus on devising an improved SLE method that would detect the pedestrian's steps and subsequently estimate the step length in real-time by processing the accelerometer data at the device itself. Our proposed method transforms the measured acceleration values along the earth coordinate system and also applies sliding window meaning to mitigate the negative effects of the smartphone's orientation and gravitational bias on the accuracy of SD and SLE. The performances of our proposed method are evaluated in terms of accuracy for five different users by taking the device in two different postures (handheld and trouser pocket) under two different walking modes (normal and fast) to demonstrate its efficacy. Moreover, our proposed method also obtains more than 88% accuracy (median) for all participants when the device is placed in trouser pocket under two different walking modes.

Index Terms—step detection, step length, sensor, smartphone, accelerometer, machine leaning, accuracy.

I. INTRODUCTION

Pedestrian's step detection and step length estimation are crucial for designing the localization and navigation systems utilized for offering various location-based services [1, 2]. Such systems could be effectively used in applications such as, guide for blind, assisted living, emergency rescue, and so on. Satellite-signal based localization, which exploits either global navigation satellite system (GNSS) or global positioning system (GPS), works well in outdoor areas. But such localization techniques cannot provide satisfactory localization performance in indoor areas or urban canyons since direct satellite signals cannot reach those areas. In order to overcome this difficulty, many indoor localization systems based on radio frequency technologies such as, Wi-Fi, ultra-wide band (UWB), wireless sensor networks, radio frequency identification (RFID), Long Range (LoRA), and so on have been proposed in the literature [3]. Among the various localization algorithms existing in literature, time delay-based localization schemes that depend on time-of-arrival (ToA) or time-difference-of arrival (TDoA) measurements, cannot determine the user's location without knowing the positional co-ordinates of the access points (APs) or beacons [4, 5, 6]. Fingerprint based localization algorithms, on the other hand, require a database of fingerprints collected at several known...
locations within the field of localization in the offline phase and determine the user’s location in online phase by taking the location of offline fingerprint that matches best with the online fingerprint [7, 8, 9].

However, these localization systems require availability of certain infrastructure made up of the concerned RF technology in the area of localization for their proper working. Moreover, such localization systems, in general, provide the location estimate of a static user or object rather than a moving one. Low-cost inertial sensor based pedestrian dead reckoning (PDR) method that does not need availability of any infrastructure in the neighboring areas, is another powerful solution to indoor localization and navigation [10, 11]. Integration of the inertial sensors like accelerometer, gyroscope etc., with the smartphones available nowadays has motivated the researchers to develop smartphone sensor-based PDR localization system. Step detection (SD) and step length estimation (SLE), on the other hand, are the two fundamental operations of the PDR based localization systems since by adding the pedestrian’s estimated step length to his/her initial location successively, pedestrian’s current location can be easily determined. Design and implementation of an accurate SD and/or SLE method, thus, has become an important field of research.

Majority of the existing inertial sensor based SLE techniques proposed in the literature over the past decade collect data either from inertial sensors mounted on some body parts or the smartphone’s sensors and process them at some server or cloud to determine the step length [12, 13]. Very few SD and SLE methods exist in the literature that process and analyze the collected sensor data at the device itself in real-time. Another important issue which has been overlooked by majority of the researches is the negative effect of the smartphone’s orientation on the performances of SD and SLE methods with respect to the earth coordinate system (ECS). It basically depends on the placement position of the smartphone. Even though a few SLE methods have addressed the above-said issue, they either cannot provide reliable results or use some inertial sensors usually not available in most smartphones. The availability of accelerometer in every smartphone has motivated us to focus on designing an improved smartphone accelerometer-based step detection and step length estimation scheme in this paper. Our proposed method works on the sensor measurements at the device itself rather than sending them to a remote server or cloud for necessary processing. This feature of our proposed technique further extends its applicability for emergency navigation where Internet connectivity may not be ensured. The contributions of our proposed work are listed below.

- Design of a smartphone-based improved step detection and step length estimation technique to reduce both the usage of network bandwidth and transmission delay. The proposed approach works in real-time at the device itself.
- Solution based solely on the accelerometer sensor, which exists on every smartphone available nowadays.
- Mitigation of the negative effect of smartphone’s orientation on the performances of SD and SLE method by transforming raw acceleration data along the ECS.
- Adoption of a sliding window meaning based low-pass filter [14] by the proposed method to remove the gravitational component and other noises from the acceleration magnitude.
- Evaluation of the proposed system and its comparison with a contemporary technique in terms of step detection accuracy and step length estimation accuracy by placing the smartphone in two different positions (handheld or trouser pocket) under two different walking modes (normal and fast) to demonstrate the efficacy of the proposed system.

The paper is structured as follows. We review the various existing SD and SLE techniques in Section 2. Section 3 presents our proposed system. In Section 4, we evaluate and compare our proposed method in terms of accuracy by placing the smartphone in two different positions under two different walking modes. The concluding remarks along with future research direction are given in Section 5.

II. RELATED WORK

Many step detection and step length estimation techniques using either the inertial sensors attached to some body parts or smartphone’s built-in sensors have been proposed in the literature over the past few decades. State-of-the-art review of such techniques are provided in this section.

A. Step Detection Techniques

The existing SD techniques based on smartphone’s inertial sensor usually employ peak-valley detection procedure [15, 16, 29], zero-point detection procedure [17] or some machine learning model [20]. Although the peak-valley detection or zero-point detection-based SD techniques are comparatively simple and have low computational overhead, detection of proper peak-valley sequences by such techniques depends on the use of some threshold and their performances also get affected by the user’s walking pattern as well as the sensor’s placement position. To improve the SD accuracy a combination of both peak-valley detection and zero-point detection are applied by Zhang et al. in [18]. In [19], Yao et al. combines zero-crossing detection with the dynamic time wrapping based prediction of peak points for accurate detection of the step boundaries. Abiad et al. in [20], have proposed a machine-learning based step detection method that is designed to detect the steps for various types of human gait and also to work effectively irrespective of the sensor placement on human body, step mode, hand motion mode etc.

B. Step Length Estimation Techniques

Most of the inertial sensor based existing SLE techniques are based on some biomechanical model or machine learning method. In some cases, the empirical relation between the step length and sensor measurements is also used. In [21], Ho Lee et al., proposed a motion-aware step length estimation technique processing measurements collected from a smartphone’s built-in sensors. In this proposed technique, the hybrid model of decision tree (DT), artificial neural network (ANN) and support vector machine (SVM) are used to identify the
user’s motions. Moreover, map-based in-flight calibration is used in order to analyze the user’s gait characteristic as well as enhance the step length estimation. To verify the proposed algorithm various experiments have been conducted where the experimental results show the effectiveness of the proposed algorithm in computing the step length for any kind of motion. A smartphone sensors-based adaptive step length estimator that can estimate the walking distance at different walking speeds of the user is proposed by Huynh Ho et al., in [22]. The proposed method, at first, applies a fast Fourier transform (FFT)-based smoother on the collected acceleration data and then analyses them based on some step-detection rules for the purpose of detecting the walking steps. In [23], Martinek et al., applies the weighted context to estimate the step length of the pedestrians. The continuous wavelet transform analysis is used to detect the step time boundaries, whereas a relevance vector machine is used to determine the pedestrian context probabilities by the step detection algorithm proposed in [23].

Several existing SLE models require calibrating the user-specific parameter before using it for step length estimation purposes [24 - 26]. In [27], Bylemans et al., propose an SLE model to be used on the smartphone based on the assumption that regularly used devices are in general kept in pockets. The proposed SLE method considers the step duration, the difference between two consecutive peak acceleration points and the average acceleration to determine the step length. The SLE model proposed by Mikov et al. in [28] uses the step duration in the place of the calibration parameter in Weinberg’s formula in order to simplify the process. Smartphone sensor based SLE model proposed by Strozzi et al., in [29] transforms the measured acceleration into the linear acceleration before applying it to the SLE model and evaluates the performances of the proposed model under the different poses of smartphone like navigation mode, phoning mode etc. In [30], Lu et al. apply fuzzy logic to dynamically adjust the user-specific parameter of the Weinberg’s SLE model [24] for the different pedestrians. The proposed SLE method uses dynamic threshold-based peak-valley detection method and provides satisfactory results under different walking modes of various pedestrian. A few researches on smartphone-based PDR system conducted by Lee and Huang [31] and also by Tian et al. [32] aim to recognize different poses of smartphone like handheld, swinging or in the pocket and their effect on the positioning results. The smartphone sensors-based step detection and step length estimation technique proposed by Yao et al., in [19] addresses the problem of recognizing pedestrian’s different walking pattern using random forest algorithm.

A few researchers, on the other hand, have worked on inverted pendulum model to derive the step length [33 - 35]. An inverted pendulum-based gait model has been proposed to derive the trajectory of the center of mass (CoM) by Zijlstra and Hof in [33]. An empirical approximation of the same has been proposed in [34] to derive the step length by considering the changes in the vertical displacement of the CoM. By extending Zijlstra’s proposed gait model, Lueken et al., have deduced a mathematical model that relates acceleration measurements with the proposed model to estimate the step length in [35]. The SLE model proposed by Yan et al. in [36] uses deep belief network (DBN) made up of several restricted Boltzmann machines in order to learn various features of inertial sensor measurements and fit the input data based on the probability distribution. The proposed method combines zero-crossing detection of the acceleration signal with peak-valley detection of gyro signal for step detection purposes. Long short-term memory and convolutional neural network based adaptive learning is applied to retrieve the different elements for changing and recognition activities in another stride length estimation model proposed by Shu et al. in [37]. The proposed model combines the features obtained from the learning module via some suitable fusion method to determine the stride length. The SLE model recently proposed by Vezočnik and Juric in [38] considers the magnitude of accelerometer sensor and step frequency as input. The proposed SLE model provides satisfactory results when it is evaluated by walking on a treadmill as well as a test polygon with rectangular shape. Our recently proposed foot mounted inertial sensor based SLE model that processes both accelerometer and gyroscope sensor measurements to estimate step length is presented in [39].

Most of the above-mentioned SD and SLE techniques process the sensor data collected from the body-mounted inertial sensors at the remote server or cloud rather than processing them locally. Moreover, the majority of such techniques have not considered the negative effect of the smartphone’s orientation on the performances of SD and SLE methods. Thus, we aim to design an improved SD and SLE method, which would transform the collected raw acceleration data along the ECS and then further process them at the device itself to estimate the step length in this paper.

III. PROPOSED SYSTEM

The proposed smartphone-based step detection (SD) and step length estimation (SLE) technique is composed of three parts which are sensor data collection using android sensor API (phase I), data processing (phase II) and then step detection along with step length estimation using a simplified SLE model discussed in phase III. The work flow diagram for our proposed system, which is named as IRT-SD-SLE is summarized in fig. 1. The constituent parts of the proposed system are described below.

A. Phase I: Sensor Data collection

In the first phase (phase I) of our proposed system data from accelerometer and rotation vector sensor are collected in periodic manner. Android sensor application programming interface (API) provides a set of methods to collect and manage the raw data from the embedded physical sensors of a smart handheld device. The full documentation of this API is available at [40]. To collect data from some sensor of an android device, it is required to create and activate a listener over the needed sensor, which can be either an actual hardware sensor or virtual sensor, by making a call to the registerListener method. SensorManager.registerListener(this, SensorManager.getDefaultSensor(Type_sensor), S_rate), where SensorManager is the object that lets the user to access the
device’s sensors, Type_sensor indicates the identification code associated with the needed sensor type and S_rate defines the sampling rate at which data is read from the specified sensor. The virtual sensor is a software-based sensor created by android operation system and it derives data from some hardware sensor. The examples of such sensors are linear acceleration, rotation vector sensor and etc.

A localization and tracking application in general require the device’s acceleration values along the world reference system, i.e., earth co-ordinate system (ECS) for accurate estimation of the step length. The x and y axes belonging to smartphone’s body co-ordinate system (denoted by X^B and Y^B) are aligned to forward and upward directions respectively whereas z axis (denoted as Z^B) remains vertical to the phone screen as shown in fig. 2. The x, y and z axes of the ECS (denoted as X^W, Y^W, Z^W), on the other hand, points to the local east, north and up directions respectively as shown in fig. 2. To transform the acceleration values measured along the device’s body coordinate system into that corresponding to ECS, it is required to know the device’s orientation with respect to the ECS. The orientation of a device denoted by a pair of an angle and an axis, can be obtained by processing the measurements of rotation vector sensor, a kind of android virtual sensor. In the above said method, if Type_sensor is set to Sensor.TYPE_ROTATION_VECTOR, the listener method for rotation vector sensor is activated. The first three elements of the rotation vector are equal to the last three components of a unit quaternion represented as \(\langle \cos \left(\frac{\varphi}{2}\right), \ x \cdot \sin \left(\frac{\varphi}{2}\right), \ y \cdot \sin \left(\frac{\varphi}{2}\right), \ z \cdot \sin \left(\frac{\varphi}{2}\right) \rangle\), where \(\varphi\) is the angle through which the device has rotated around the axes \((x, y, z)\). The fourth element of the rotation vector is equal to the scalar component of the unit quaternion, i.e., \(\cos \left(\frac{\varphi}{2}\right)\).

**B. Phase II: Data Processing and Transformation**

The second phase (phase II) of our proposed system processes the collected sensor data and then transforms the acceleration measurements into the corresponding values along the ECS. To transform the raw acceleration data collected from the hardware sensor into the corresponding acceleration values along the ECS, it is required to compute the orientational or rotation matrix by processing the orientation measurements collected from the rotation vector virtual sensor. At first, four elements of unit quaternion \(Q = [u_0 \ u_1 \ u_2 \ u_3]\) is obtained from the values of rotation vector sensor (denoted as RV) as follows.

\[
\begin{align*}
    u_0 &= \cos \left(\frac{\varphi}{2}\right) = RV[3] \\
    u_1 &= x \cdot \sin \left(\frac{\varphi}{2}\right) = RV[0] \\
    u_2 &= y \cdot \sin \left(\frac{\varphi}{2}\right) = RV[1] \\
    u_3 &= z \cdot \sin \left(\frac{\varphi}{2}\right) = RV[2]
\end{align*}
\]

The orientational or rotation matrix at ith time slot is computed as follows.

\[
O(i) = \begin{bmatrix}
1 - 2(u_0^2 + u_2^2) & 2(u_1u_2 - u_0u_3) & 2(u_1u_3 + u_0u_2) \\
2(u_1u_2 + u_0u_3) & 1 - 2(u_1^2 + u_2^2) & 2(u_2u_3 - u_0u_1) \\
2(u_1u_3 - u_0u_2) & 2(u_2u_3 + u_0u_1) & 1 - 2(u_1^2 + u_2^2)
\end{bmatrix}
\]

Now, the raw acceleration measured at ith time slot along the sensor body framework (denoted as \(Acc^B(i)\)) can be transformed to the same along the world reference system (denoted as \(Acc^W(i)\)) by applying the following equation.

\[
Acc^W(i) = Acc^B(i) \cdot O(i)
\]

**C. Phase III: Step Detection and Step Length Estimation**

The third phase of our proposed system (phase III) aims to detect the steps taken by the pedestrian and then estimate
the corresponding step length in real time as shown in fig. 1. Our proposed step length estimation (SLE) technique uses, at first, a sliding window meaning based low-pass filter and then a peak-valley detection procedure to detect the steps of the pedestrian as well as estimate the corresponding step lengths. Various sub procedures of our proposed SLE technique are described below.

1) Removal of gravitational component from acceleration magnitude via low-pass filter: The magnitude of transformed 3-axis acceleration data (denoted as \( \text{Acc}^m \)) at \( i \)th time slot is computed as follows.

\[
\text{Acc}^m(i) = \sqrt{(\text{Acc}_x^m(i))^2 + (\text{Acc}_y^m(i))^2 + (\text{Acc}_z^m(i))^2} \tag{4}
\]

where \( \text{Acc}_x^m(i) \), \( \text{Acc}_y^m(i) \) and \( \text{Acc}_z^m(i) \) denotes transformed acceleration values along the \( X \), \( Y \) and \( Z \)-axis respectively at \( i \)th time slot. Since the force of gravity has an influence on the measured acceleration, thus it must be eliminated from the acceleration magnitude to obtain the actual acceleration values. To eliminate the gravitational component and other noise factors from the acceleration magnitude, our proposed SLE technique uses sliding window meaning (SWM) based low-pass filter [14]. SWM based low-pass filter adopted by our proposed method uses the following equations to remove the contribution of gravitational force and other noises from the acceleration magnitude.

\[
\text{Acc}_f^m(i) = \|\text{Acc}^m(i) - \text{Acc}^m\|, \\
\text{Acc}^m = \frac{1}{n} \sum_{i=1}^{n} \text{Acc}^m(i), \tag{5}
\]

where \( n \) is the window size. On the other hand, by subtracting the gravity (\( g = 9.8 \)) component from the transformed \( z \)-axis acceleration, linear \( z \)-axis acceleration component (denoted as \( \text{Acc}^{lin}_z \)) at the \( i \)th sample is computed as follows.

\[
\text{Acc}^{lin}_z = \text{Acc}^m_z - g \tag{6}
\]

Fig. 3 shows the variation of both linear vertical acceleration component, i.e., acceleration \( z \)-axis component and filtered acceleration magnitude with the data samples in range of 400 to 800.

2) Detection of steps using peak valley identification procedure: According to the researchers in [13], each step cycle consists of exactly one maximum or peak point followed by exactly one minimum point or valley point. To identify both maximum points and minimum points from the filtered acceleration magnitude, our proposed technique have considered two different thresholds which are peak threshold (\( \text{th}_\text{peak} \)) and valley threshold (\( \text{th}_\text{valley} \)) as shown by the flow chart given in fig. 1. To determine the valid peak points from the set of identified maximum points, some restrictions on the interval between two consecutive peak points (denoted by the variable \( \text{min}_\text{peak, interval} \)) in terms of samples are imposed by our proposed method. Moreover, to identify the step cycles using a valid pair of peak and valley points, another condition variable named as \( \text{det}_\text{peak} \) is introduced. The value of condition variable \( \text{det}_\text{peak} \) is initialized to false and it becomes true upon detecting a valid peak point when the following conditions are met.

\[
\begin{align*}
C1 & : \text{Acc}_f^m(i) > \text{th}_\text{peak} \\
C2 & : i - \text{LP1} > \text{min}_\text{peak, interval} \\
C3 & : \text{det}_\text{peak} = \text{false};
\end{align*}
\]

Here, \( i \) and \( \text{LP1} \) denotes current sample index and last peak index respectively. \( \text{LP1} \), which is initialized to 0, is set to current sample index whenever a valid peak point is detected. On the other hand, valid valley point is found when the following conditions are satisfied.

\[
\begin{align*}
C1 & : \text{Acc}_f^m(i) > \text{th}_\text{valley} \\
C2 & : \text{det}_\text{peak} = \text{true};
\end{align*}
\]

The identification of the step cycles using the valid pairs of peak valley sequence within the range of data samples from
TABLE I

<table>
<thead>
<tr>
<th>Sl No</th>
<th>Height</th>
<th>Weight</th>
<th>Sex</th>
<th>BMI</th>
<th>β</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>170</td>
<td>80</td>
<td>M</td>
<td>27.7</td>
<td>0.38</td>
</tr>
<tr>
<td>2</td>
<td>145</td>
<td>55</td>
<td>F</td>
<td>26.2</td>
<td>0.33</td>
</tr>
<tr>
<td>3</td>
<td>170</td>
<td>74</td>
<td>M</td>
<td>25.6</td>
<td>0.38</td>
</tr>
<tr>
<td>4</td>
<td>168</td>
<td>86</td>
<td>M</td>
<td>30.5</td>
<td>0.28</td>
</tr>
<tr>
<td>5</td>
<td>155</td>
<td>84</td>
<td>F</td>
<td>35</td>
<td>0.32</td>
</tr>
</tbody>
</table>

220 to 720 is depicted by fig. 4. The values of peak threshold (th_peak) and valley threshold (th_valley) is set to 2.0 and -1.5 respectively as shown in fig. 4, whereas min_peak_interval is set to 15 based on the reasoning that the pedestrians can take at most three steps per second [41].

3) Estimation of step length using Weinberg Model: Following the Weinberg’s proposed SLE model, step length (SL) of the pedestrian is estimated by the following equation.

\[
ESL_j = \beta \cdot \sqrt{\text{Acc}_{z,\text{max}}(j) - \text{Acc}_{z,\text{min}}(j)}
\]  

where \( ESL_j \) denotes the user’s estimated step length at the \( j \)th step, whereas \( \text{Acc}_{z,\text{max}}(j) \) and \( \text{Acc}_{z,\text{min}}(j) \) denote the maximum and minimum linear vertical acceleration within the \( j \)th step cycle respectively. \( \beta \), which is a user dependent parameter, is required to be calibrated for each individual pedestrian. The value of \( \beta \) can be determined by using the measured values of maximum and minimum linear vertical acceleration as well as the actual step length in the following equation.

\[
\beta = \frac{\text{Actual SL}}{\sqrt{\text{Acc}_{z,\text{max}}(j) - \text{Acc}_{z,\text{min}}(j)}}
\]  

IV. EXPERIMENTAL RESULTS AND ANALYSIS

We have used Galaxy M13 5G smartphone having 2.2 GHz octa-core processor for experimentation purposes. Since calibration of parameter \( \beta \) is required to be done for different users before evaluation of our proposed method, so parameter \( \beta \) is calibrated at first for five different participants.

A. Calibration of user parameter

A linear distance of 5 meters was walked by five different participants to calibrate \( \beta \). Based on the calculated average step length (which is obtained by dividing the traveled distance by the number of foot steps taken by the pedestrian) along with the maximum and minimum vertical acceleration within each step cycle of individual pedestrian, the value of \( \beta \) is estimated by using equation 8. The calibrated values of \( \beta \) for five pedestrians are provided in table 1. The relationship between the parameter \( \beta \) and pedestrian’s body mass index (BMI) is depicted in fig. 5. Such relationship between \( \beta \) and user’s BMI would help to predict the value of \( \beta \) for any new user without the user’s participation in the calibration process.

V. PERFORMANCE ANALYSIS

The performances of our proposed technique IRT-SD-SLE have been evaluated and compared with another existing technique proposed by Strozzi et al. [29] in terms of step detection (SD) accuracy and step length estimation (SLE) accuracy. A linear distance of 12 meters has been walked by five participants in four different cases that are resulted by placing the device in two different postures (handheld and trouser pocket) under two different walking modes (normal and fast) for analyzing the performances of the proposed technique. The linear path within the experimental site walked by some participant while keeping the smartphone in handheld mode, is shown in fig. 6. Equations 9 and 10 define the SD accuracy and SLE accuracy, respectively.

\[
\text{SD Accuracy} = \left(1 - \frac{|DS - AS|}{AS}\right) \times 100%,
\]

where \( DS \) and \( AS \) indicate the number of steps detected by SD method and the actual number of steps taken by the user respectively.
Fig. 7. Screen shot of android app designed for Real-time step detection and step length estimation based on our proposed method.

TABLE II
ANALYSIS OF SD ACCURACY WHEN DEVICE IS HELD IN HAND UNDER NORMAL WALKING MODE

<table>
<thead>
<tr>
<th>User Id</th>
<th>Actual Steps</th>
<th>Steps detected by [29]</th>
<th>Steps det. by IRT-SD-SLE</th>
<th>Accuracy for [29]</th>
<th>Accuracy for IRT-SD-SLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>21</td>
<td>15</td>
<td>17</td>
<td>71.43%</td>
<td>80.95%</td>
</tr>
<tr>
<td>2</td>
<td>23</td>
<td>18</td>
<td>20</td>
<td>78.26%</td>
<td>86.96%</td>
</tr>
<tr>
<td>3</td>
<td>18</td>
<td>13</td>
<td>19</td>
<td>72.22%</td>
<td>94.44%</td>
</tr>
<tr>
<td>4</td>
<td>21</td>
<td>18</td>
<td>20</td>
<td>93.24%</td>
<td>85.71%</td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td>15</td>
<td>17</td>
<td>75.00%</td>
<td>85.00%</td>
</tr>
</tbody>
</table>

pedestrian respectively.

\[ SLE \text{ Accuracy} = \left(1 - \frac{|ED - AD|}{AD}\right) \times 100\%, \tag{10} \]

where \(ED\) denotes the estimated distance provided by the proposed technique and \(AD\) denotes actual distance travelled by the user. The value of \(AD\) in our case is 12 meters, whereas the estimated distance is obtained by successively adding the step length estimated by the SLE technique for each step taken by the user during the experimental walk.

The experimental results in terms of SD accuracy for five different participant pedestrians in four different cases are given in tables II, IV, VI and VIII respectively. The same for five different pedestrians in four different cases in terms of SLE accuracy are provided in tables III, V, VII and IX respectively. The accuracy results in term of both step detection and step length estimation given in tables II-IX

TABLE III
ANALYSIS OF SLE ACCURACY WHEN DEVICE IS HELD IN HAND UNDER NORMAL WALKING MODE

<table>
<thead>
<tr>
<th>User Id</th>
<th>Actual dist.(m)</th>
<th>Dist.(m) estimated by [29]</th>
<th>Dist.(m) est. by IRT-SD-SLE</th>
<th>Accuracy for [29]</th>
<th>Accuracy for IRT-SD-SLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>11.50</td>
<td>8.21</td>
<td>9.83</td>
<td>71.39%</td>
<td>85.48%</td>
</tr>
<tr>
<td>2</td>
<td>12.00</td>
<td>9.47</td>
<td>10.01</td>
<td>78.92%</td>
<td>83.42%</td>
</tr>
<tr>
<td>3</td>
<td>12.00</td>
<td>7.11</td>
<td>11.30</td>
<td>59.25%</td>
<td>94.20%</td>
</tr>
<tr>
<td>4</td>
<td>12.00</td>
<td>11.19</td>
<td>10.11</td>
<td>93.25%</td>
<td>84.25%</td>
</tr>
<tr>
<td>5</td>
<td>12.00</td>
<td>8.21</td>
<td>9.02</td>
<td>68.42%</td>
<td>80.17%</td>
</tr>
</tbody>
</table>

TABLE IV
ANALYSIS OF SD ACCURACY WHEN DEVICE IS HELD IN POCKET (TROUSER) UNDER NORMAL WALKING MODE

<table>
<thead>
<tr>
<th>User Id</th>
<th>Actual Steps</th>
<th>Steps detected by [29]</th>
<th>Steps det. by IRT-SD-SLE</th>
<th>Accuracy for [29]</th>
<th>Accuracy for IRT-SD-SLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>21</td>
<td>16</td>
<td>20</td>
<td>76.19%</td>
<td>95.24%</td>
</tr>
<tr>
<td>2</td>
<td>24</td>
<td>19</td>
<td>21</td>
<td>79.17%</td>
<td>87.50%</td>
</tr>
<tr>
<td>3</td>
<td>17</td>
<td>13</td>
<td>17</td>
<td>76.47%</td>
<td>100.00%</td>
</tr>
<tr>
<td>4</td>
<td>21</td>
<td>18</td>
<td>23</td>
<td>85.71%</td>
<td>90.48%</td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td>15</td>
<td>18</td>
<td>75.00%</td>
<td>90.00%</td>
</tr>
</tbody>
</table>

TABLE V
ANALYSIS OF SLE ACCURACY WHEN DEVICE IS HELD IN POCKET (TROUSER) UNDER NORMAL WALKING MODE

<table>
<thead>
<tr>
<th>User Id</th>
<th>Actual dist.(m)</th>
<th>Dist.(m) estimated by [29]</th>
<th>Dist.(m) est. by IRT-SD-SLE</th>
<th>Accuracy for [29]</th>
<th>Accuracy for IRT-SD-SLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12.00</td>
<td>8.73</td>
<td>11.32</td>
<td>72.75%</td>
<td>94.36%</td>
</tr>
<tr>
<td>2</td>
<td>12.00</td>
<td>9.39</td>
<td>10.61</td>
<td>78.25%</td>
<td>88.32%</td>
</tr>
<tr>
<td>3</td>
<td>12.00</td>
<td>6.78</td>
<td>10.38</td>
<td>56.50%</td>
<td>86.53%</td>
</tr>
<tr>
<td>4</td>
<td>12.00</td>
<td>10.35</td>
<td>11.97</td>
<td>86.25%</td>
<td>99.77%</td>
</tr>
<tr>
<td>5</td>
<td>12.00</td>
<td>8.09</td>
<td>10.31</td>
<td>67.42%</td>
<td>85.92%</td>
</tr>
</tbody>
</table>

TABLE VI
ANALYSIS OF SD ACCURACY WHEN DEVICE IS HELD IN HAND UNDER FAST WALKING MODE

<table>
<thead>
<tr>
<th>User Id</th>
<th>Actual Steps</th>
<th>Steps detected by [29]</th>
<th>Steps det. by IRT-SD-SLE</th>
<th>Accuracy for [29]</th>
<th>Accuracy for IRT-SD-SLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>19</td>
<td>13</td>
<td>15</td>
<td>68.42%</td>
<td>78.35%</td>
</tr>
<tr>
<td>2</td>
<td>21</td>
<td>14</td>
<td>17</td>
<td>66.67%</td>
<td>80.95%</td>
</tr>
<tr>
<td>3</td>
<td>16</td>
<td>13</td>
<td>19</td>
<td>81.25%</td>
<td>81.25%</td>
</tr>
<tr>
<td>4</td>
<td>17</td>
<td>11</td>
<td>15</td>
<td>64.71%</td>
<td>88.24%</td>
</tr>
<tr>
<td>5</td>
<td>18</td>
<td>13</td>
<td>15</td>
<td>72.22%</td>
<td>83.33%</td>
</tr>
</tbody>
</table>

TABLE VII
ANALYSIS OF SLE ACCURACY WHEN DEVICE IS HELD IN HAND UNDER FAST WALKING MODE

<table>
<thead>
<tr>
<th>User Id</th>
<th>Actual dist.(m)</th>
<th>Dist.(m) estimated by [29]</th>
<th>Dist.(m) est. by IRT-SD-SLE</th>
<th>Accuracy for [29]</th>
<th>Accuracy for IRT-SD-SLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12.00</td>
<td>7.05</td>
<td>9.41</td>
<td>58.75%</td>
<td>78.42%</td>
</tr>
<tr>
<td>2</td>
<td>12.00</td>
<td>7.33</td>
<td>10.21</td>
<td>61.08%</td>
<td>85.08%</td>
</tr>
<tr>
<td>3</td>
<td>12.00</td>
<td>7.83</td>
<td>11.56</td>
<td>65.25%</td>
<td>96.31%</td>
</tr>
<tr>
<td>4</td>
<td>12.00</td>
<td>6.06</td>
<td>9.53</td>
<td>50.50%</td>
<td>79.42%</td>
</tr>
<tr>
<td>5</td>
<td>12.00</td>
<td>7.62</td>
<td>9.27</td>
<td>63.50%</td>
<td>77.25%</td>
</tr>
</tbody>
</table>
show that our proposed technique IRT-SD-SLE achieves better accuracy compared to the existing method proposed in [29] in all four cases. Moreover, the proposed technique IRT-SD-SLE achieves better results in terms of both SD and SLE accuracy when the phone is placed in pocket (trouser) compared to the device’s handheld position under two different walking modes. This happens because the device’s orientation changes more rapidly when it is held in hand compared to when it is placed in trouser pocket. On the other hand, SD and SLE accuracies obtained by our proposed technique for all five participants are more than 88% (median) when the device is placed in pocket under the normal walking mode and the accuracies fall slightly when the user walks fast due to the above-said reason.

The screenshot of the android app designed for real-time step detection and step length estimation purposes based on our proposed method IRT-SD-SLE is provided in fig. 7.

### VI. CONCLUDING REMARKS AND FUTURE RESEARCH DIRECTIONS

A smartphone accelerometer based real-time step detection and step length estimation technique is proposed in this paper. Our proposed technique can detect the pedestrian’s steps and subsequently estimate his/her step length in real-time by processing only the accelerometer measurements at the device itself to reduce the network bandwidth and transmission delay. Moreover, our proposed method attempts to mitigate the negative effect of smartphone’s orientation on the SD and SLE accuracy by transforming raw acceleration data along the ECG by using rotation matrix and also adopts a sliding window meaning based low pass filter to remove the gravitational component and other noises from the acceleration magnitude. The proposed technique employs a peak valley procedure that uses two thresholds and also imposes a constraint on the interval between consecutive peak points to avoid over detection of the step cycles. The performance of our proposed method is evaluated by holding the device in two most commonly used different postures under two different walking modes. The performance analysis in terms of SD accuracy and SLE accuracy for five different users show that our proposed technique achieves better results in terms of both accuracies compared to the existing one considered here. Furthermore, the proposed method obtains more than 85% accuracy (median) in terms of step detection and step length estimation for all five participants when the device is placed in trouser pocket under two different walking modes (normal and fast).

However, the performances of our proposed technique have been evaluated by walking a linear path only while the smartphone is held in two different positions. So, we intend to evaluate its performance by taking a curvaceous walk that includes either left or right turn as well as considering several other device postures in near future. Moreover, our proposed method cannot achieve desirable accuracy when the device is held in hand under fast walking mode. We, thus, plan to resolve this issue also in future.

### REFERENCES


[39] Bibhas Gayen, Pampa Sadhukhan, Chandreyee Chowdhury et al. Human Gait Modeling with Step Length Estimation based on Single Foot Mounted Inertial Sensors, 03 February 2023, PREPRINT (Version 1) available at Research Square [https://doi.org/10.21203/rs.3.rs-2262623/v1]
