



Sensors (Basel). 2016 Sep; 16(9): 1423.

PMCID: PMC5038701

Published online 2016 Sep 2. doi: [10.3390/s16091423](https://doi.org/10.3390/s16091423)

PMID: [27598171](https://pubmed.ncbi.nlm.nih.gov/27598171/)

Step-Detection and Adaptive Step-Length Estimation for Pedestrian Dead-Reckoning at Various Walking Speeds Using a Smartphone

Ngoc-Huynh Ho, Phuc Huu Truong, and Gu-Min Jeong^{*}

Angelo Maria Sabatini, Academic Editor

School of Electrical Engineering, Kookmin University, 861-1 Jeongnung-dong, Seongbuk-gu, Seoul 136-702, Korea; ngochuynh@kookmin.ac.kr (N.-H.H.); phtruong@kookmin.ac.kr (P.H.T.)

^{*}Correspondence: gm1004@kookmin.ac.kr; Tel.: +82-2-910-4408

Received 2016 Apr 12; Accepted 2016 Aug 29.

Copyright © 2016 by the authors; licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).

Abstract

We propose a walking distance estimation method based on an adaptive step-length estimator at various walking speeds using a smartphone. First, we apply a fast Fourier transform (FFT)-based smoother on the acceleration data collected by the smartphone to remove the interference signals. Then, we analyze these data using a set of step-detection rules in order to detect walking steps. Using an adaptive estimator, which is based on a model of average step speed, we accurately obtain the walking step length. To evaluate the accuracy of the proposed method, we examine the distance estimation for four different distances and three speed levels. The experimental results show that the proposed method significantly outperforms conventional estimation methods in terms of accuracy.

Keywords: Weinberg method, PDR, step length, step-detection, FFT filter

1. Introduction

Pedestrian dead-reckoning (PDR) is extensively studied as an effective approach to obtain pedestrian locations by estimating the distance traveled via handheld inertial sensors [1]. This method can be developed at a low cost for use in conjunction with new services. With the rapid development of microelectromechanical systems (MEMS) in recent years, the demand for pedestrian positioning has been considerably increasing in fields such as navigation systems [2], augmented reality [3], gait analysis [4], and health monitoring [5]. Therefore, it is necessary to accurately detect human walking distances for the determination of pedestrian locations.

Recently, two main types of algorithms for distance estimation have been described. The first type is based on the successive double-integral-based length-step measurement of acceleration. The major drawback of this technique is the error accumulated over the duration of the experiments. This problem can be partially addressed using zero velocity updates (ZUPT). In the second type, researchers applied a verifiable relationship between vertical acceleration and the step length to estimate the distance traveled by a moving subject. Bylemans et al. [6] constructed an empirical solution for step-length estimation using mobile devices equipped on the subject's body. Another empirical method was based on the correlation of vertical acceleration at the foot of the subject [7,8,9] with the length of the step. Wang et al. [7] developed a wearable sensor module that could be attached to the ankles of the subjects in order to collect the acceleration signal during different walking activities, and then to estimate the walking

distance. Indeed, these conventional methods showed good performance in the case of normal walking speed. However, if the user's speed increased, the estimation errors became excessively large.

Considering these facts, we propose a step-length estimation method based on a previously developed method (namely, the Weinberg method [10]) to improve the estimation accuracy under three speed levels: low, normal, and high. To this end, we define a unit conversion as a function of the step velocity and determine the step length using the Weinberg formula. In the proposed method, we apply the fast Fourier transform (FFT) operation to smooth the obtained accelerations before identifying the step-phase using our step-detection rules. Furthermore, we transform the sensor coordinate system to the world coordinate system for step-length estimation. After the step-detection process, we estimate the step velocity using a linear regression model, and thus determine the length of step. Conclusively, we sum all step lengths during traveling to obtain the walking distance.

The remainder of this paper is structured as follows. First, a review of the conventional methods is presented in [Section 2](#). [Section 3](#) provides our definition of the device's coordinate system and the ranges of the step levels and presents our proposed method for walking-distance estimation. [Section 4](#) presents the experimental results for four healthy subjects. Finally, the conclusions are presented in [Section 5](#).

2. Related Works

In a PDR system, to update the pedestrian's position, it is necessary to first estimate the length of each step. Numerous PDR estimators were previously proposed for the estimation of the walking distance both indoors [11,12,13,14] and outdoors [14,15]. One of the most renowned distance estimators for smartphones based on a PDR algorithm was presented by Weinberg [10]. According to this method, vertical acceleration was applied to the subject's hip, in order to determine the walking distance. Then, a unit conversion was used to adjust the estimated walking distance. The proposed Weinberg formula is:

$$L_1 \approx K_1 \times \sqrt[4]{A_{max} - A_{min}} \quad (1)$$

where A_{max} and A_{min} are the maximum and minimum acceleration values, respectively, measured on the Z-axis in a single stride, respectively, and K_1 is a constant for unit conversion (i.e., feet or meters traveled).

Kim [16] developed an empirical method based on the dependence of average acceleration with each step length during walking. The step length using this method is:

$$L_2 = K_2 \times \sqrt[3]{\frac{\sum_{i=1}^M |a_i|}{N}} \quad (2)$$

where a_i is the acceleration measured on sample i -th in a single step, M is the number of samples corresponding to each step, and K_2 is a constant for unit conversion.

Finally, Tian [14] designed a tracking system for a handheld device in multi-mode. This method estimates the travel distance based on the height of the subjects and the step frequency. This method's formula can be expressed as:

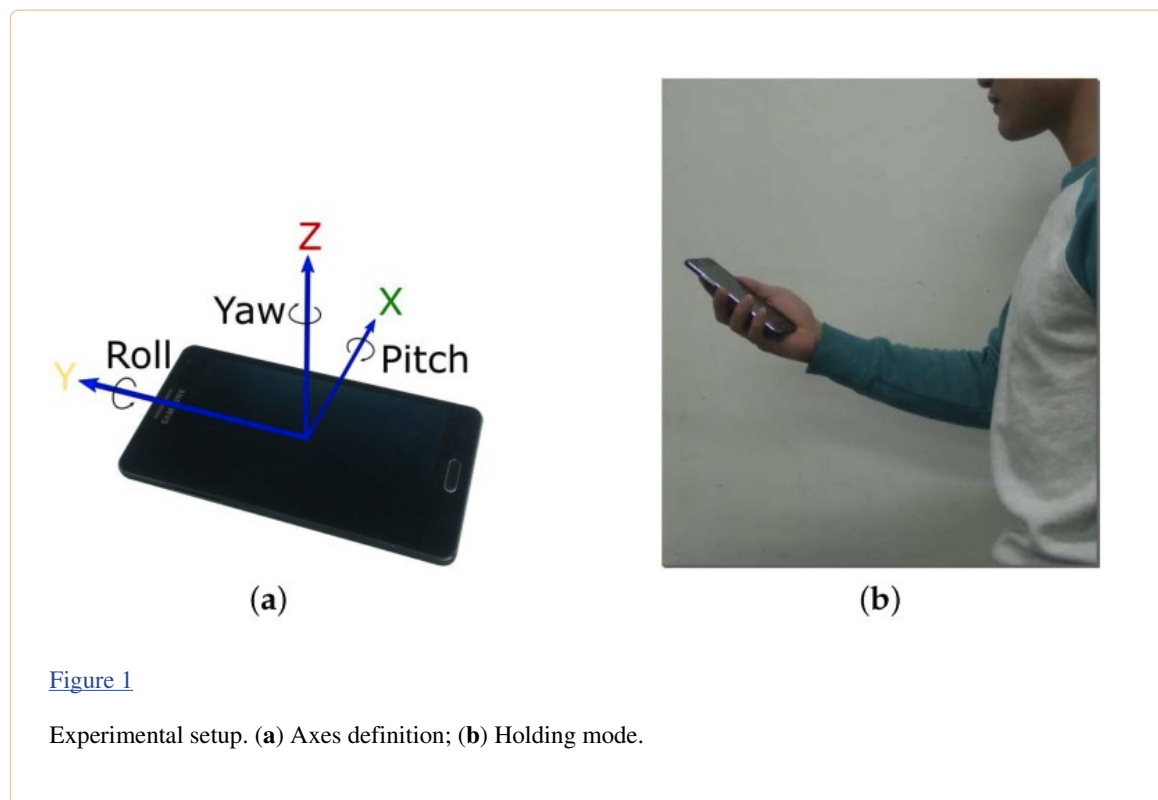
$$L_3 = K_3 \times h \times \sqrt{f_s} \quad (3)$$

where f_s is the step frequency measured during the walking experiment, h is the height of a certain subject, and K_3 is a constant for unit conversion.

3. Walking Distance Estimation Algorithm

3.1. Experimental Setup and Speed Level Definition

We developed an Android application to collect motion data during the experiments. The lateral direction, longitudinal, and perpendicular directions were denoted as x -axis, y -axis, and z -axis, respectively, as shown in [Figure 1a](#). The experimental setup is illustrated in [Figure 1b](#). Throughout the experiment, the users held the phone in their hand in front of their chest. The participants (four males and four females) were required to walk straight ahead for four different distances: 10, 20, 30, and 40 m. In order to extrapolate the experimental results to different pedestrian groups, the experiment was repeated at three speed levels: low, normal, and high.



Knoblauch et al. [17] determined the walking speed of young pedestrians (ages 14 to 64) at 1.25 m/s and the walking speed of older pedestrians (ages 65 and older) at 0.97 m/s. According to Bohannon et al. [18], the mean comfortable speed was 1.27 m/s for women and 1.46 m/s for men, whereas the mean maximum speed was 1.75 m/s for women and 2.53 m/s for men. Obaidi et al. [19] concluded that the mean minimum walking speed is 0.8 m/s. It ranges from 1.08 to 1.24 m/s for normal walking, and it reaches 1.81 m/s for fast walking. Tarawneh et al. [20] evaluated the pedestrian speed in Jordan, and found that pedestrian speeds vary from 1.17 to 1.49 m/s. Furthermore, pedestrians between 21 and 30 years old were the fastest group, whereas, those over 65 years were the slowest group. Chandra et al. [21] proposed an average walking speed from 0.97 to 1.36 m/s on various types of facilities (e.g., sidewalk, wide-sidewalk, and precinct). Based on these findings, we defined the speed levels to be used in the present experiments from a normal distribution of average walking velocities as shown in [Figure 2](#). Specifically, we determined the ranges of low, normal, and high speed as $v \leq \bar{v} - \sigma$, $\bar{v} - \sigma < v < \bar{v} + \sigma$, and $v \geq \bar{v} + \sigma$, respectively. Here, \bar{v} and σ are the average speed and deviation of human walking speed, respectively. From [Figure 2](#), we set $\bar{v} = 1.36$ m/s and $\sigma = 0.29$. Thus, these walking speed ranges are [0.68, 1.07], [1.08, 1.64], and [1.65, 2.05], respectively, where 0.68 and 2.05 are the minimum and maximum walking speeds in our experiments.

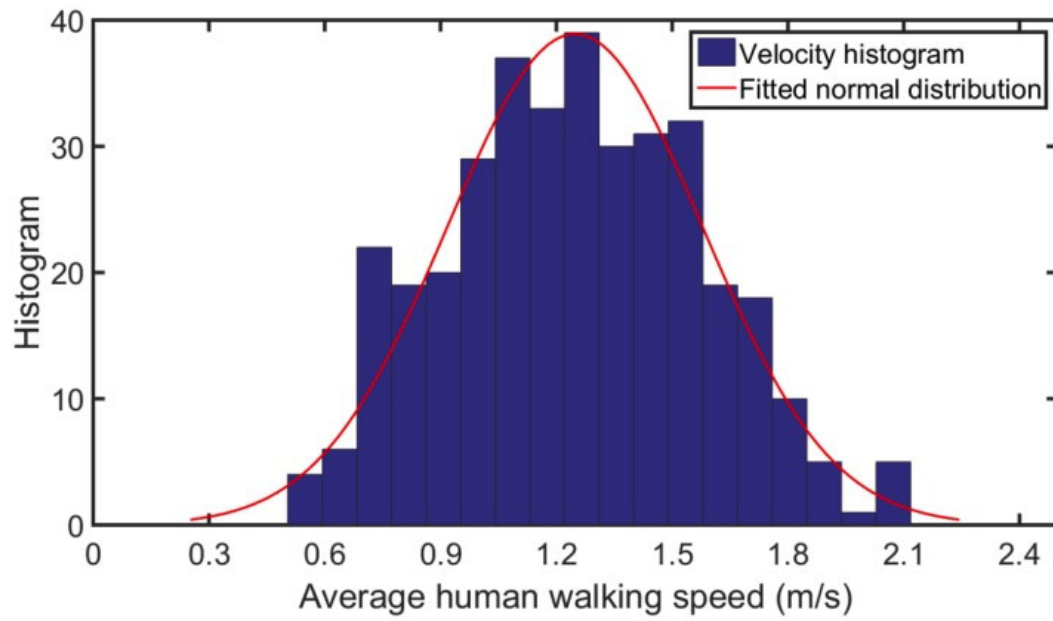


Figure 2

Distribution of average walking speed.

3.2. FFT Operation-Based Data Smoothing

We apply FFT [22] to smooth the raw signals for step-detection. Each acceleration signal is represented by a four-dimensional function $f(X, Y, Z, t)$, where (X, Y, Z) denotes the space axes, whereas t denotes the time axis. We define the first-order finite differences operator matrix (\mathbf{D}) as follows:

$$\mathbf{D}(\mathbf{f}) = \text{vec}(f(X, Y, Z, t + 1) - f(X, Y, Z, t)) \quad (4)$$

where \mathbf{f} and $\text{vec}(\cdot)$ represent the vectorized version of the space–time function $f(X, Y, Z, t)$ and the vectorization operator, respectively. Our purpose is to obtain a smoothed signal after the inverse FFT (IFFT) process. Therefore, it is necessary to solve error by minimizing the problem in [22]. Then, we obtain the smoothed acceleration function \mathbf{A} as follows:

$$\mathbf{A} = \mathbf{F}^{-1} \left[\frac{\mathbf{F} [\mu \times \mathbf{a} + \rho \times \mathbf{D}^T (\mathbf{u}) - \mathbf{D}^T (\mathbf{y})]}{\mu + \rho \times \mathbf{F} [|\mathbf{D}|]^2} \right] \quad (5)$$

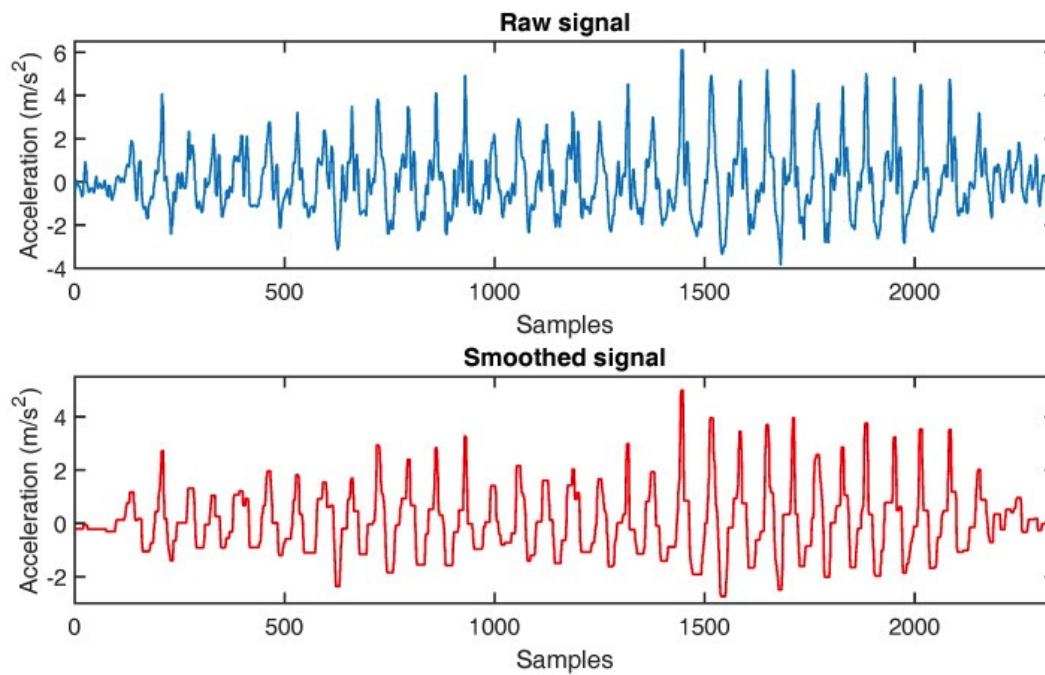
where $\mathbf{F}[\cdot]$ and $\mathbf{F}^{-1}[\cdot]$ are denoted for the FFT and inverse fast Fourier transform (IFFT) functions, respectively. \mathbf{D}^T is the transpose of \mathbf{D} , and μ, ρ are the input and regularization parameters, respectively. To solve the intermediate variable error, letting $\mathbf{v} = \mathbf{D}(\mathbf{f}) + \frac{1}{\rho} \times \mathbf{y}$, we obtain \mathbf{u} as follows:

$$\mathbf{u} = \max \left\{ |\mathbf{v}| - \frac{1}{\rho}, 0 \right\} \times \text{sign}(\mathbf{v}) \quad (6)$$

In addition, the Lagrange multiplier \mathbf{y} was obtained from [22] as follows:

$$\mathbf{y}_{l+1} = \mathbf{y}_l - \rho \times (\mathbf{u}_{l+1} - \mathbf{D}(\mathbf{A}_{l+1})) \quad (7)$$

Algorithm 1 summarizes the FFT operation-based data smoothing method by pseudo-code. [Figure 3](#) shows an example of a raw signal and the corresponding smoothed signal on the Z-axis. This figure clarifies that the smoothed signal is less fluctuant than the raw signal. Therefore, we use the smoothed signal to detect the walking step.



[Figure 3](#)

Smoothing of Z-axis acceleration.

Algorithm 1 Pseudo code for FFT operation-based data smoothing method

```

Input data: a
Input parameter:  $\mu$ 
Set parameter:  $\rho = 10, l = 100$ 
Initialize  $\mathbf{f}_0 = 0, \mathbf{u}_0 = 0, \mathbf{y}_0 = 0$ 
Compute the finite differences operator matrix  $\mathbf{D}$  and its transpose  $\mathbf{D}^T$ 
for  $i = 1:l$ 
  1. Solve  $\mathbf{A}$  using Equation (5)
  2. Solve  $\mathbf{u}$  using Equation (6)
  3. Update the Lagrange multiplier  $\mathbf{y}$  using Equation (7)
end

```

3.3. Step Detection Rules

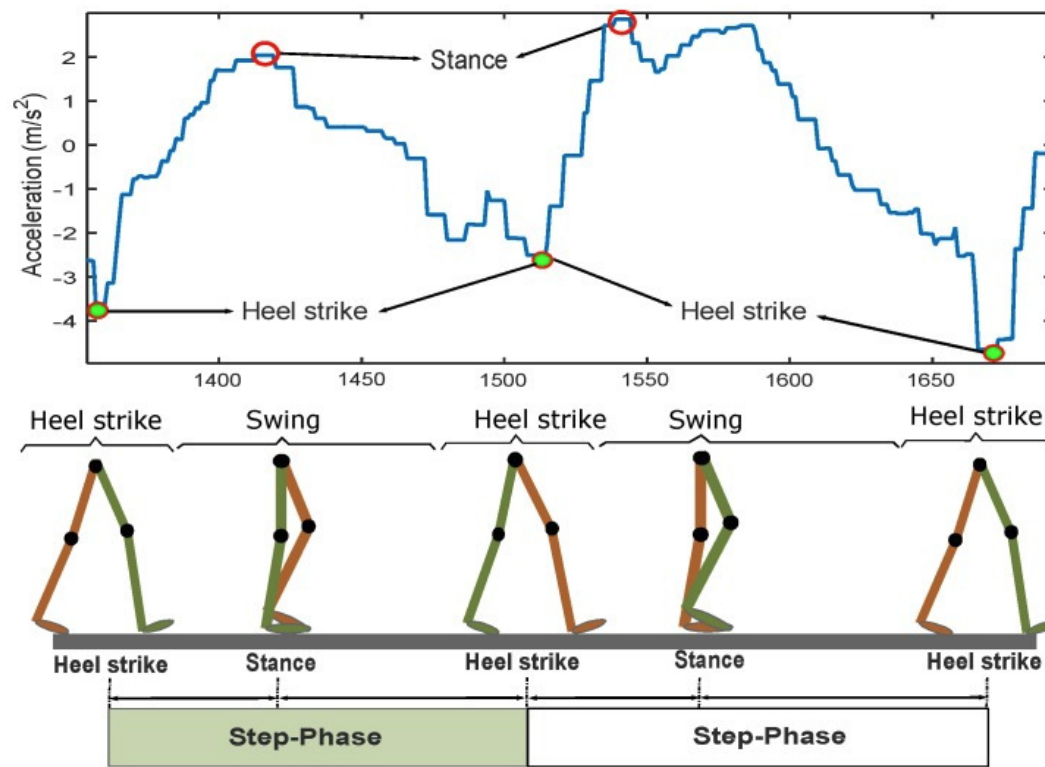
Step detection is an important problem in PDR that is calculated using the summation of all estimated step lengths. Because of false or missed detection, it can cause substantial errors in total walking distance estimation. Therefore, it is necessary to correctly detect the occurring-step moment. Recently, step detection methods using accelerometers were presented in PDR investigation. Three common methods for acceleration data processing were developed. The first method [23] analyzed the difference of the acceleration data along the vertical dimension (Z-axis) while the user was walking. The disadvantage of this method is that the acceleration data could be confusing if the orientation of the accelerometer changed. In the second method [24], the root mean square (RMS) of three axes was used

to analyze the feature of the step. The RMS concept used for processing the raw acceleration data is calculated as follows:

$$RMS = \sqrt{a_X^2 + a_Y^2 + a_Z^2} \quad (8)$$

where a_X , a_Y , and a_Z are the acceleration data of the X, Y, and Z axes, respectively. However, the acceleration sensor did not always maintain a steady state when the user was walking. Therefore, noise tended to increase proportionally with the pedestrian moving speed. The last method is a combination of the acceleration and rotation matrices. In [25], authors used the rotation matrices to transform the acceleration data from the sensor frame to the Earth frame. This method focused on the acceleration data along the vertical dimension. Although the gesture of the accelerometer changed, the data along the vertical orientation maintained normal. Thus, compared with the first method, this method could be more useful in a non-steady state during the movement of the user. However, this method is only valid for low speed movements.

In our experiments, the smartphone was tightly held in the users' hands while they were walking straight ahead. Thus, we can state that the orientation did not significantly change. Based on this analysis, we applied the first method for step detection without considering its drawback. First, we define that a step is composed of a swing phase and a heel strike phase [26]. The swing phase occurs when the reference foot moves forward from the behind of the contralateral leg. The heel strike phase happens when the heel contacts the ground and the waist (the center of gravity) is in its lowest position during the entire step. In the swing phase, the vertical acceleration begins from a minimum valley, then goes to a maximum peak, and finally stops at another minimum valley, which is caused by the next heel strike phase. In other words, the vertical acceleration reaches one maximum peak and one minimum valley within a step. [Figure 4](#) illustrates the pattern of the vertical acceleration in a step.



[Figure 4](#)

Vertical acceleration based on walking diagram.

To detect steps, determination of the maximum peaks (maxima) and minimum valleys (minima) of acceleration data using thresholds can be used. However, this method does not effectively work due to the large variation of the amplitudes of the minima during the heel strike phase. This problem causes no maximum or two maxima between two consecutive minima. On the contrary, the amplitudes of the maxima in the swing phase is less variable. Therefore, it is easy to detect the maxima, but difficult to detect the minima. Considering these facts, we design a new method to fit the minima set. The flow of this method is described in [Figure 5](#).

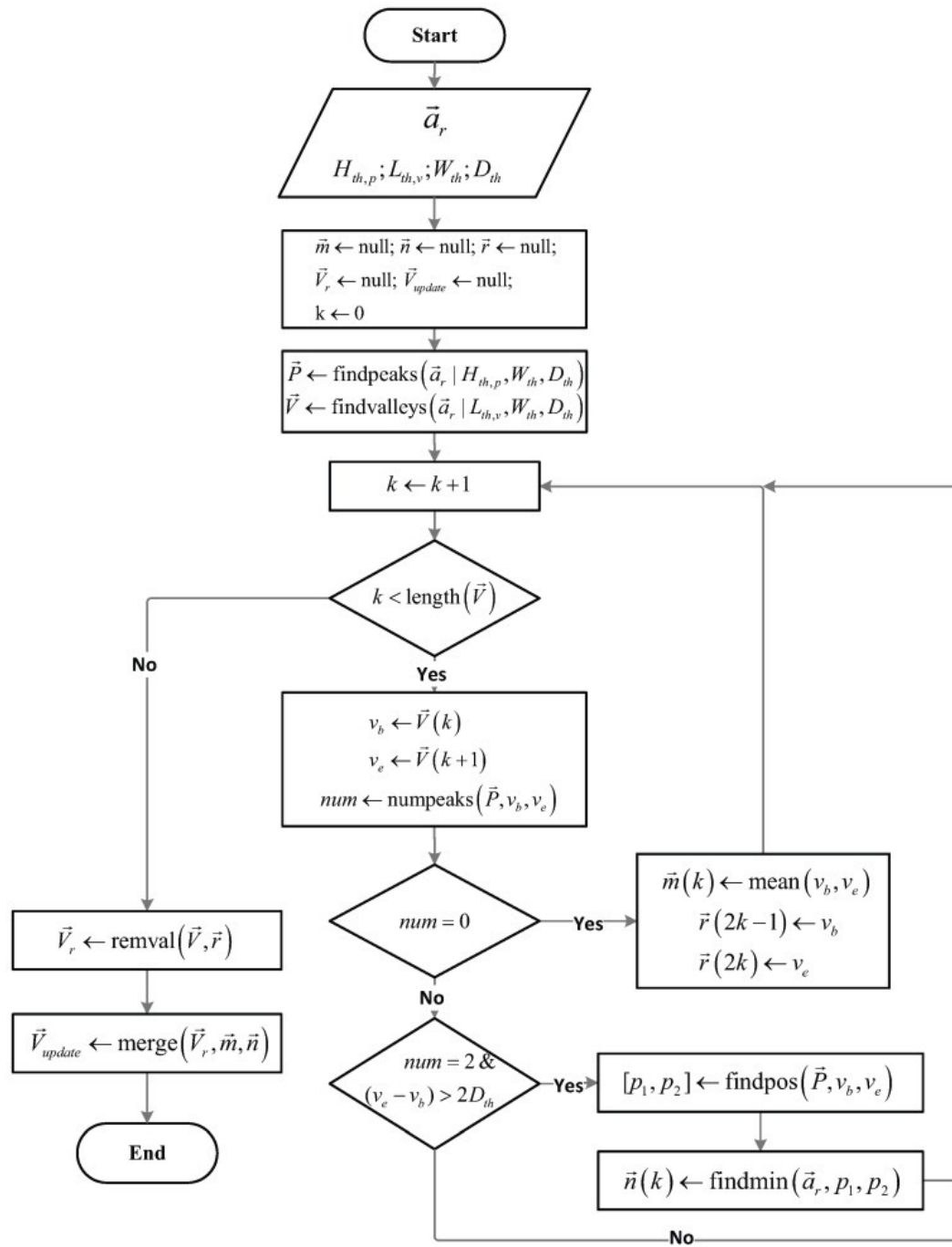


Figure 5

Flowchart of minima-adjustment method.

First, we obtain the smoothed-acceleration data \vec{a}_r using FFT smoother, and assign four thresholds: minimum-peak height ($H_{th,p}$), maximum-valley height ($L_{th,v}$), minimum width (W_{th}), and minimum distance (D_{th}). The minimum-peak height is used to find peaks greater than $H_{th,p}$, whereas the maximum-valley height is utilized to detect the valleys smaller than $L_{th,v}$. We use the minimum width to locate peaks or valleys whose widths are at least W_{th} . The minimum distance is used to find peaks or valleys where the distances between two peaks or two valleys are longer than D_{th} . Moreover, we define \vec{m} , \vec{n} , and \vec{r} as the vectors containing the regulated (\vec{m} and \vec{n}) and removed (\vec{r}) minima. Based on the thresholds, we determine two sets of maxima \vec{P} and minima \vec{V} using two functions $\text{findpeaks}(\bullet)$ and $\text{findvalleys}(\bullet)$, respectively. Then, using the function $\text{numpeaks}(\bullet)$, we calculate the number of maxima between two consecutive minima v_b and v_e . If there is no maximum in the interval between v_b and v_e ,

we remove those minima and define a new minimum at the middle of the two removed minima using the function $\text{mean}(\bullet)$, as shown in Figure 6a. The positions of the removed and regulated minima are stored in \vec{r} and \vec{m} , respectively. If there are two maxima between two consecutive minima and the distance of these minima is greater than twice of D_{th} , we determine p_1 and p_2 positions of those maxima using the function $\text{findpos}(\bullet)$. In Figure 6b, we locate the lowest valley between p_1 and p_2 , and set it as a new minimum using the function $\text{findmin}(\bullet)$. We store the position of the new minima in \vec{n} . Then, we remove the unexpected minima in \vec{r} by applying the function $\text{remval}(\bullet)$ to obtain a new minima set, \vec{V}_r . Finally, using the function $\text{merge}(\bullet)$, we obtain the updated minima set \vec{V}_{update} from \vec{V}_r , \vec{m} , and \vec{n} for detection of the step-window and to count the number of steps.

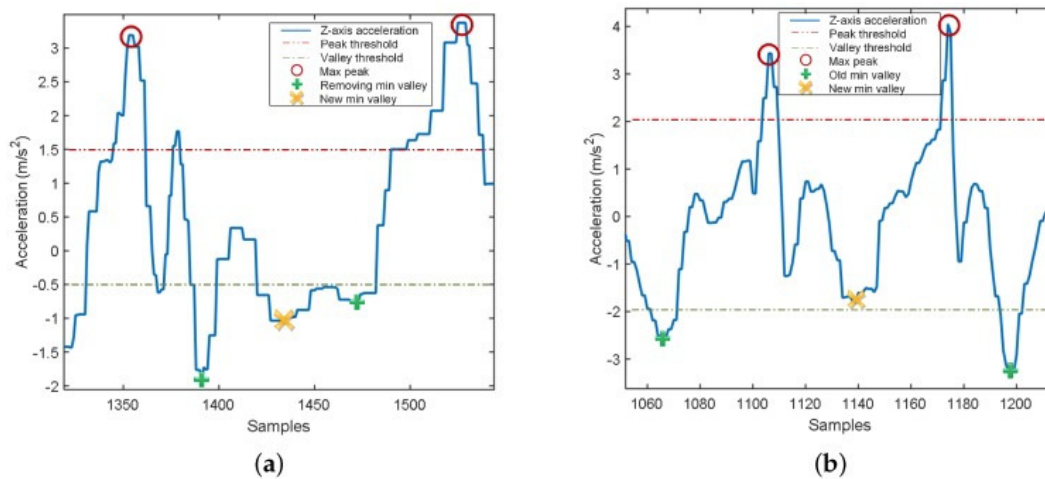


Figure 6

Unexpected minima-set solution in specific cases: (a) No maximum; (b) Two maxima.

Figure 7 shows an example of the maxima and minima detection and a step counting result after detecting step windows. We determined the step-sliding window for each step, where the stopping point of the specified window corresponded to the starting point of the next window. We defined the starting point of the first window and the stopping point of the last window as the zero acceleration preceding the first maxima and the zero acceleration succeeding the last maxima, respectively. The duration between two consecutive red dashed vertical lines in Figure 7 indicates the step window size. We detected 33 windows corresponding to 33 walking steps during user movement.

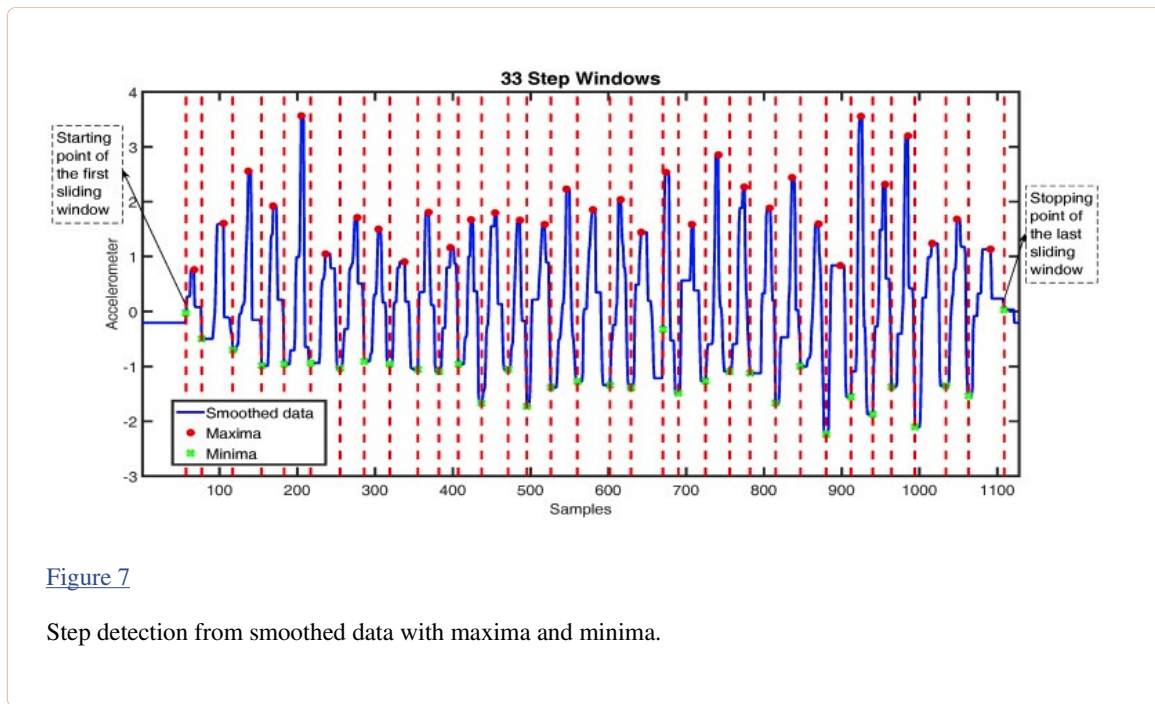


Figure 7

Step detection from smoothed data with maxima and minima.

3.4. Coordinate System Transformation Using Euler's Rotation Theorem

Generally, the collected acceleration data are in the smartphone's coordinates; thus, it is necessary to transform the movement data to the world coordinates. We transformed the three-axis accelerations from the smartphone's coordinates to the world coordinates using Euler's rotation theorem [27]. First, we constructed the acceleration and angle vectors from the raw data as follows:

$$\vec{V}_{acc} = \begin{bmatrix} a_X \\ a_Y \\ a_Z \end{bmatrix} ; \vec{V}_{angle} = \begin{bmatrix} \phi \\ \psi \\ \theta \end{bmatrix} \quad (9)$$

where, ϕ , ψ , and θ are the yaw, the roll, and the pitch of the phone's symmetrical axes, respectively. We calculated $T_c = \cos(\vec{V}_{angle})$ and $T_s = \sin(\vec{V}_{angle})$. Suppose that $T_c = [cx \ cy \ cz]^T$ and $T_s = [sx \ sy \ sz]^T$, the rotation matrix could be constructed by Euler angles to the Z-Y-X convention as follows:

$$\mathbf{R}_{ZYX} = \begin{bmatrix} cy \times cz & sy \times sx \times cz - sz \times cx & sy \times cx \times cz + sz \times sx \\ cy \times sz & sy \times sx \times sz + cz \times cx & sy \times cx \times sz - cz \times sx \\ -sy & cy \times sx & cy \times cx \end{bmatrix} \quad (10)$$

Multiplying the rotation matrix with the acceleration vector, we obtained the rotated acceleration vector as follows:

$$\begin{bmatrix} a_{rX} \\ a_{rY} \\ a_{rZ} \end{bmatrix} = \mathbf{R}_{ZYX} \times \vec{V}_{acc} \quad (11)$$

where, a_{rX} , a_{rY} , and a_{rZ} are the rotated acceleration data on three dimensional axes of the world

coordinate system.

3.5. Adaptive Step Length Based on Unit Conversion

In the proposed method, instead of using a constant K for all steps, we derived a K -factor as a polynomial function of the average step velocity. To choose the degree of the polynomial function, we applied a N -fold Cross-Validation on the dataset. First, we randomly divided the dataset into two sets: a training set and a test set, where the ratio between training set and data adapts from 0.2 to 0.9, and the root mean square errors (RMSE) of the actual velocity and estimated velocity are shown in [Figure 8a](#). From this figure, we decided that the test set included 30% of the samples, whereas the training set composes 70% of the total samples. We examined the variance of the residual sum of squares (vRSS) with k first degrees of the polynomial function and repeated this procedure 100 times ($N = 100$). Then, we calculated the mean value of the variances and chose the optimal degrees that minimized the variance in the test set. [Figure 8b](#) shows the vRSS for the degree selection of the polynomial function in the test set and the training set. From this figure, we chose two first degrees, \bar{v}_{step} and \bar{v}_{step}^2 , to train the K -value using a linear regression model. The adaptive K -value was obtained as follows:

$$K_{vel} = 0.68 - 0.37 \times \bar{v}_{step} + 0.15 \times \bar{v}_{step}^2 \quad (12)$$

where \bar{v}_{step} was computed as the magnitude of the average velocities on three dimensional axes, X , Y , and Z in each step:

$$\bar{v}_{step} = \sqrt{\bar{v}_{stepX}^2 + \bar{v}_{stepY}^2 + \bar{v}_{stepZ}^2} \quad (13)$$

where \bar{v}_{stepX} , \bar{v}_{stepY} , and \bar{v}_{stepZ} were obtained using the accelerometer double-integral formulas:

$$\begin{aligned} \bar{v}_{stepX} &= \text{mean} \left(\int a_{r_stepX}(t) dt \right) \\ \bar{v}_{stepY} &= \text{mean} \left(\int a_{r_stepY}(t) dt \right) \\ \bar{v}_{stepZ} &= \text{mean} \left(\int a_{r_stepZ}(t) dt \right) \end{aligned} \quad (14)$$

where a_{r_stepX} , a_{r_stepY} , and a_{r_stepZ} were the acceleration on three dimensional axes of the world coordinate system. The concept of acceleration transformation from the device coordinate system to the world coordinate system is described in [Section 3.4](#). Substituting Equation (12) into Equation (1), for all cases of low, normal, and high speed, we obtained the step length formula as:

$$L_{step} = K_{vel} \times \sqrt[4]{A_{max} - A_{min}} \quad (15)$$

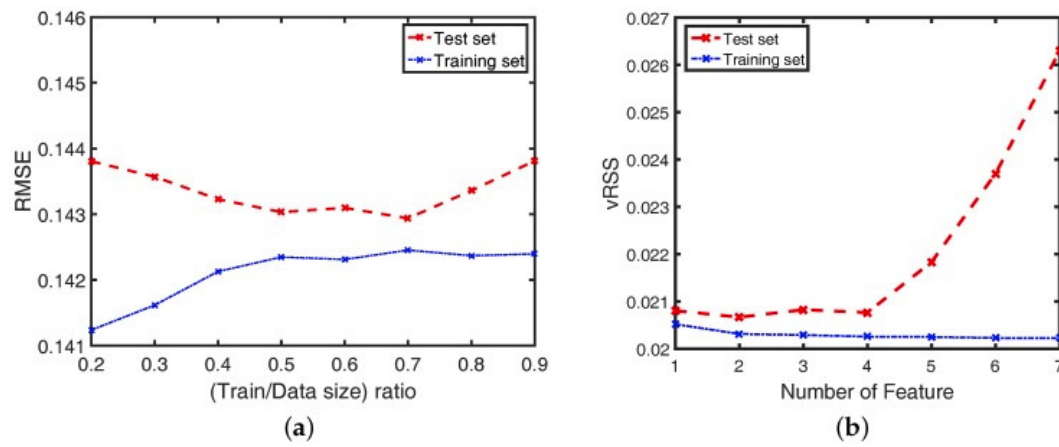


Figure 8

(a) RMSE of the actual and estimated velocities for training-set and test-set separation; (b) Variance of residual sum of squares (RSS) for degree decision of the polynomial function.

Denoting N as the number of steps walked during the experiment, we established the traveled distance D by summing all the adaptive step lengths as follows:

$$D = \sum_{i=1}^N L_{step}(i) \quad (16)$$

4. Experimental Results

In our experiments, we conducted the above-mentioned walking experiments on eight participants (four males and four females). During this experiment, each person conducted walking experiments covering four distances (10, 20, 30, and 40 m) at three different speed levels (low, normal, and high). The range of walking speeds estimated using Equation (13) was from 0.68 to 2.05 m/s. Figure 9 shows a comparison of the walking distance estimators in the cases of low, normal, and high walking speeds. The vertical axis corresponds to the estimated distance. To illustrate estimating results, we plot the maximum and minimum values, the lower and upper quartiles, and the median of the estimated distance for each estimator. Our proposed estimator provided a better performance than other methods in all speed levels, where the mean estimated distance (the red line) of the proposed method approximates the actual distance (20 m).

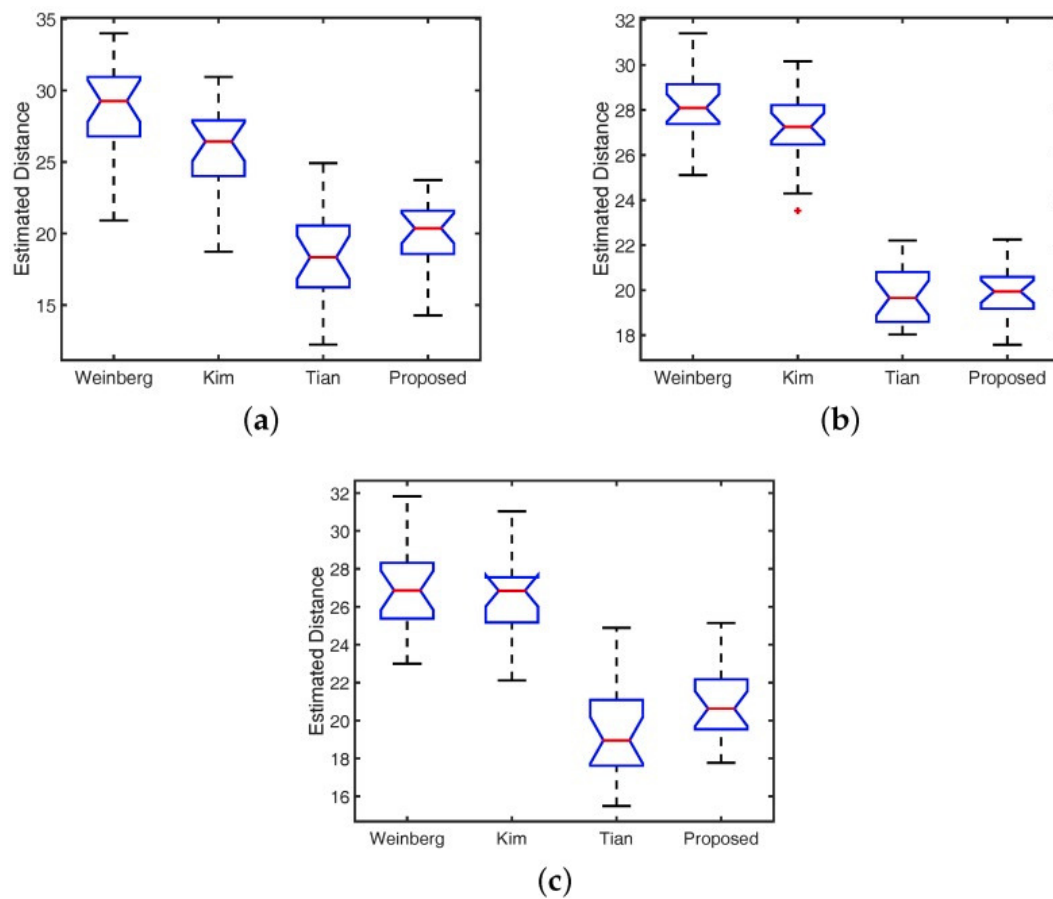


Figure 9

Comparison of the distance estimation methods. (a) Slow Walking; (b) Normal Walking; (c) Fast Walking.

Table 1 shows the averages of the estimated walking speeds and step lengths of the proposed method for the four travel distances covered at the three different walking-speed levels. In addition, the table presents the standard deviation (Std) of the average velocity in each distance at each speed level. The last column presents the average results of four indicated distances. Based on these results, the users' step lengths increase as their speed increases.

Table 1

Speed information and step-length estimation.

Speed Level		10 m	20 m	30 m	40 m	Average
Low	Mean \pm Std (m/s)	0.94 ± 0.06	0.93 ± 0.04	0.93 ± 0.03	0.95 ± 0.02	0.937 ± 0.040
	Step length (m)	0.57	0.60	0.62	0.59	0.595
Normal	Mean \pm Std (m/s)	1.35 ± 0.03	1.36 ± 0.04	1.37 ± 0.04	1.36 ± 0.04	1.360 ± 0.037
	Step length (m)	0.70	0.67	0.70	0.69	0.690
High	Mean \pm Std (m/s)	1.72 ± 0.11	1.70 ± 0.04	1.69 ± 0.04	1.69 ± 0.04	1.700 ± 0.065
	Step length (m)	0.81	0.82	0.78	0.78	0.797

[Table 2](#) shows the estimation error (in percentage) and the standard deviation (in meters) results for each walking distance at each speed level of four estimators. The average error column presents the mean error of three speed levels. The results indicate that in all cases, the proposed method performed better than the reference methods. Particularly, Weinberg [10] and Kim [16] methods acquired high error rates with an average of about 20%, whereas, Tian [14] method obtained better results, where the error rate was about 13%. The proposed method significantly reduced the distance estimation error to 5%. Unlike those methods [10,14,16] which use a constant K -value in all the steps for all experimental observations, the proposed method used a velocity-based K -value estimator to determine the correlative step length. Therefore, the proposed method can obtain better accuracy of distance estimation for various speed levels.

Table 2

Estimation errors and standard deviations of walking distance estimators.

Method	Distance (m)	Low Speed		Normal Speed		High Speed		Average Error (%)
		Error (%)	Std (m)	Error (%)	Std (m)	Error (%)	Std (m)	
Weinberg [10]	10	16.95	1.16	17.94	0.69	24.03	0.67	19.64
	20	21.44	1.87	20.67	1.72	25.62	1.27	22.57
	30	16.71	2.45	21.56	1.97	26.42	1.91	21.56
	40	19.00	2.64	22.01	2.81	27.61	2.45	22.87
Kim et al. [16]	10	22.59	1.22	18.54	0.74	20.17	0.82	20.43
	20	26.05	1.87	20.53	2.03	22.55	1.42	23.04
	30	22.10	2.01	21.55	2.57	23.30	2.50	22.31
	40	23.42	2.56	22.32	3.13	24.73	2.84	23.69
Tian et al. [14]	10	14.36	1.91	10.86	0.99	7.25	0.91	10.82
	20	15.48	3.64	8.16	2.42	10.53	2.39	11.24
	30	11.34	3.60	11.03	3.12	11.74	3.45	11.37
	40	19.48	4.83	13.15	3.26	20.13	6.72	17.58
Proposed Method	10	4.76	1.31	4.44	0.87	5.06	1.13	4.89
	20	4.51	1.86	4.78	1.84	4.15	1.69	4.48
	30	4.52	2.91	4.51	2.41	4.63	2.43	4.55
	40	4.32	2.15	4.46	3.22	3.81	2.94	4.19

5. Conclusions

This paper proposes a walking-distance estimation method for PDR. In particular, we introduced a new step detection algorithm and a step length estimator. By estimating the step velocity, we defined the unit conversion for each step phase in the step-length estimation process. Then, we determined the total distance during walking by summing all the step lengths. This technique improved the performance of the distance estimator for pedestrian navigation. In the near future, we will investigate the case of walking with a smartphone, such as pedestrians holding the smartphone in their hands and swinging their arms, or putting the smartphone in their pocket during walking. Moreover, we plan to test the estimation method for other types of locomotion, such as running or jogging.

Acknowledgments

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2015R1D1A1A01060917), and also supported by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government (MSIP) (NRF-2016R1A5A1012966).

Author Contributions

Authors contributed equally to this work. All authors read and approved the manuscript.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Mikov A., Moschevikin A., Fedorov A., Sikora A. A Localization System Using Inertial Measurement Units from Wireless Commercial Hand-Held Devices; Proceedings of the International Conference on Indoor Positioning and Indoor Navigation; Montbéliard-Belfort, France. 28–31 October 2013.
2. Jirawimut R., Ptasiński P., Garaj V., Cecelja F., Balachandran W. A Method for Dead Reckoning Parameter Correction in Pedestrian Navigation System. *IEEE Trans. Instrum. Meas.* 2003;52:209–215. doi: 10.1109/TIM.2002.807986. [[CrossRef](#)]
3. Kim J., Jun H. Vision-based Location Positioning Using Augmented Reality for Indoor Navigation. *IEEE Trans. Consum. Electron.* 2008;54:954–962. doi: 10.1109/TCE.2008.4637573. [[CrossRef](#)]
4. Wren T.A., Gorton G.E., Ounpuu S., Tucker C.A. Efficacy of Clinical Gait Analysis: A Systematic Review. *Gait Posture.* 2011;34:149–153. doi: 10.1016/j.gaitpost.2011.03.027. [[PubMed](#)] [[CrossRef](#)]
5. González-Valenzuela S., Chen M., Leung V.C.M. Mobility Support for Health Monitoring at Home Using Wearable Sensors. *IEEE Trans. Inf. Technol. Biomed.* 2011;15:539–549. doi: 10.1109/TITB.2010.2104326. [[PubMed](#)] [[CrossRef](#)]
6. Bylemans I., Weyn M., Klepal M. Mobile Phone-Based Displacement Estimation for Opportunistic Localisation Systems; Proceedings of the Third International Conference on Mobile Ubiquitous Computing, Services and Technologies (UBICOMM '09); Sliema, Malta. 11–16 October 2009; pp. 113–118.
7. Wang J.S., Lin C.W., Yang Y.T., Ho Y.J. Walking Pattern Classification and Walking Distance Estimation Algorithms Using Gait Phase Information. *IEEE Trans. Biomed. Eng.* 2012;59:2884–2892. doi: 10.1109/TBME.2012.2212245. [[PubMed](#)] [[CrossRef](#)]
8. Chuckpaiwong B., Nunley J.A., Mall N.A., Queen R.M. The Effect of Foot Type on In-Shoe Plantar Pressure during Walking and Running. *Gait Posture.* 2008;28:405–411. doi: 10.1016/j.gaitpost.2008.01.012. [[PubMed](#)] [[CrossRef](#)]
9. Alvarez J.C., González R.C., Alvarez D., López A.M., Rodríguez-Uría J. Multisensor Approach to Walking Distance Estimation with Foot Inertial Sensing; Proceedings of the 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, IEEE Engineering in Medicine and Biology Society; Lyon, France. 23–26 August 2007. [[PubMed](#)]
10. Weinberg H. Using the ADXL202 in Pedometer and Personal Navigation Applications. Analog Devices, Inc.; Norwood, MA, USA: 2002.
11. Jimenez A.R., Seco F., Prieto C., Guevara J. A Comparison of Pedestrian Dead-Reckoning Algorithms Using a Low-Cost MEMS IMU; Proceedings of the IEEE International Symposium on Intelligent Signal Processing (WISP); Budapest, Hungary. 26–28 August 2009.

12. Abdulrahim K., Hide C., Moore T., Hill C. Aiding MEMS IMU with Building Heading for Indoor Pedestrian Navigation; Proceedings of the Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS); Kirkkonummi, Finland. 14–15 October 2010; pp. 1–6.
13. Alvarez D., Gonzalez R.C., Lopez A., Alvarez J.C. Comparison of Step Length Estimators from Wearable Accelerometer Devices; Proceedings of the IEEE Engineering in Medicine and Biology Society; New York, NY, USA. 30 August–3 September 2006; pp. 5964–5967. [[PubMed](#)]
14. Tian Q., Salcic Z., Wang K.I.-K., Pan Y. A Multi-Mode Dead Reckoning System for Pedestrian Tracking Using Smartphones. *IEEE Sens. J.* 2016;16:2079–2093. doi: 10.1109/JSEN.2015.2510364. [[CrossRef](#)]
15. Jahn J. Comparison and Evaluation of Acceleration Based Step Length Estimators for Handheld Devices; Proceedings of the International Conference on Indoor Position and Indoor Navigation; Zurich, Switzerland. 15–17 September 2010.
16. Kim J.W., Jang H.J., Hwang D.-H., Parket C. A Step, Stride and Heading Determination for the Pedestrian Navigation System. *Positioning.* 2004;1:8. doi: 10.5081/jgps.3.1.273. [[CrossRef](#)]
17. Knoblauch R., Pietrucha M., Nitzburg M. Field Studies of Pedestrian Walking Speed and Start-up Time. *J. Transp. Res. Board.* 1996;27:27–38. doi: 10.3141/1538-04. [[CrossRef](#)]
18. Bohannon R.W. Comfortable and Maximum Walking Speed of Adults Aged 20–79 Years: Reference Values and Determinants. *Age Ageing.* 1997;26:15–19. doi: 10.1093/ageing/26.1.15. [[PubMed](#)] [[CrossRef](#)]
19. Al-Obaidi S., Wall J.C., Al-Yaqoub A., Al-Ghanim M. Basic Gait Parameters: A Comparison of Reference Data for Normal Subjects 20 to 29 Years of Age from Kuwait and Scandinavia. *J. Rehabil. Res. Dev.* 2003;40:361–366. doi: 10.1682/JRRD.2003.07.0361. [[PubMed](#)] [[CrossRef](#)]
20. Tarawneh M.S. Evaluation of Pedestrian Speed in Jordan with Investigation of Some Contributing Factors. *J. Saf. Res.* 2001;32:229–236. doi: 10.1016/S0022-4375(01)00046-9. [[CrossRef](#)]
21. Chandra S., Bharti A.K. Speed Distribution Curves for Pedestrians During Walking and Crossing. *Procedia Soc. Behav. Sci.* 2013;104:660–667. doi: 10.1016/j.sbspro.2013.11.160. [[CrossRef](#)]
22. Chan S.H., Khoshabeh R., Gibson K.B., Gill P.E., Nguyen T.Q. An Augmented Lagrangian Method for Total Variation Video Restoration. *IEEE Trans. Image Process.* 2011;20:3097–3111. doi: 10.1109/TIP.2011.2158229. [[PubMed](#)] [[CrossRef](#)]
23. Shin S.H., Lee M.S., Park C.G. Pedestrian Dead Reckoning System with Phone Location Awareness Algorithm; Proceedings of the IEEE/ION Position Location and Navigation Symposium; Indian Wells, CA, USA. 4–6 May 2010.
24. Gusenbauer D., Isert C., Krosche J. Self-Contained Indoor Positioning on off-the-Shelf Mobile Devices; Proceedings of the International Conference on Indoor Positioning and Indoor Navigation; Zurich, Switzerland. 15–17 September 2010.
25. Shin S.H., Park C.G. Adaptive Step Length Estimation Algorithm Using Optimal Parameters and Movement Status Awareness. *Med. Eng. Phys.* 2011;33:1064–1071. doi: 10.1016/j.medengphy.2011.04.009. [[PubMed](#)] [[CrossRef](#)]
26. Lan K.-C., Shih W.-Y. Using Smart-Phones and Floor Plans for Indoor Location Tracking- Withdrawn. *IEEE Trans. Hum.-Mach. Syst.* 2014;44:211–221.
27. Palais B., Palais R., Rodi S. A Disorienting Look at Euler’s Theorem on the Axis of a Rotation. *Am. Math. Mon.* 2009;116:892–909. doi: 10.4169/000298909X477014. [[CrossRef](#)]

