UMOISP: Usage Mode and Orientation Invariant Smartphone Pedometer

Arun Kumar Siddanahalli Ninge Gowda, Swarna Ravindra Babu, and Dhineshkumar Chandra Sekaran

***Abstract*— Currently available pedometer applications on smartphones use either accelerometer or gyroscope sensors to calculate the number of steps walked. The main challenge with such individual sensor based approaches is that the accuracy can be impaired due to overlap in patterns when the phone is held in different modes, namely, hand (with and without swinging) and shirt pocket or pant pocket. This paper proposes a novel approach of pedometer implementation by combining both accelerometer and gyroscope sensors. By combining these two sensors and deriving features from the raw data, the proposed system can estimate step counts more accurately in all the smartphone usage modes. The proposed pedometer is also invariant to the orientation of the smartphone in each of the usage modes. This paper also proposes a novel magnetometer-based random motion detection algorithm, which can mitigate false step counts caused by random motions during phone handling. The performance of the proposed system is tested with different users across various walking conditions, and the results show an overall step count accuracy of 98.73% across all the smartphone usage modes.**

***Index Terms*— Pedometer, phone mode identification, random motion detection, step detection.**

I. Introduction

PEDOMETERS count the number of steps taken by a user and provide an estimate of the distance walked. There exists a wide array of applications based on pedometers like Fitness trackers, Personal activity trackers, Pedestrian navigation systems, 3D virtual tour and many more. Currently available pedometers come in different forms like standalone fitness bands (Nike Fuel Band, Fitbit Charge, Samsung Gear Fit), or as applications (S Health [1], Google Fit [2], Noom, Moves) on smartphones and smart watches, or as dedicated portable wearable devices (Omron pedometers) or as smart insoles (Digitsole Tracker, [3]) for footwear.

As smartphones become more powerful and versatile with faster processor and multitude of sensors present in them, software pedometer applications based on MEMS (Micro-electromechanical systems) sensors are gaining popularity. Pedometer implementation on smartphones is quite challenging as they will have to work in all the different modes the mobile device can be carried by a user, unlike the dedicated pedometers which are designed to work at a specific wearable location. Incorrect estimate of step count, which in-turn leads to distance errors [4] can deteriorate the performance of applications dependent on pedometers, causing position errors in pedestrian navigation applications [5] and inconsistent image stitching in virtual tour applications [6]. Similarly step count errors can lead to incorrect calorie estimation in fitness tracker applications [7].

Existing MEMS sensor based pedometer applications normally use either the accelerometer or gyroscope sensor. They work by counting the number of cycles or patterns generated whenever a step is taken by the user. Among these pedometers, the ones which are based on accelerometer sensor (henceforth referred to as Accelerometer Pedometer or AP) are more common, where peak detection and pattern recognition methods are normally used for step detection. Peak detection methods rely on identifying the peaks in the accelerometer data that correspond to either the toe-off or heel strike phase of a step [8]–[10]. On the other hand zero crossing methods [4], [11], maxima and minima identification methods with time and amplitude checks [12], [13], feature extraction and classification methods [14], and finite state machine based methods [15] are used in existing works for pattern recognition based step detection. In peak detection based systems, the methods to obtain a peak threshold value for consistent step detection across different walking speeds and walking surfaces conditions (slopes) are quite challenging [10]. Some of the existing peak detection [8], [9] and pattern recognition methods [16], [17] also need a mechanism to identify and track the orientation of the device, so that acceleration data along the direction vertical to walking can be used for step detection. Pattern detection methods based on maxima and minima estimation over a fixed window of acceleration data are not adaptable to varying step frequency conditions and device usage modes. Step detection is not instantaneous in feature extraction based approaches; and a training data set is also needed to identify the various parameters that define the features for step detection. State machine based approaches perform better in step detection across limited device usage modes, but are complex to develop [15].

Several gyroscope sensor based pedometers (henceforth referred to as Gyroscope Pedometer or GP) have also been developed, where pattern recognition methods are used for

Manuscript received August 2, 2016; revised October 27, 2016; accepted November 16, 2016. Date of publication December 5, 2016; date of current version January 6, 2017. The associate editor coordinating the review of this paper and approving it for publication was Prof. Thilo Sauter.

A. K. Siddanahalli Ninge Gowda is with the Samsung R&D Institute India-Bangalore Pvt. Ltd., Bengaluru 560037, India (e-mail: arunkumar.sn@samsung.com).

S. R. Babu was with the Samsung R&D Institute India-Bangalore Pvt. Ltd., Bangalore 560037, India. He is now with Coovum Technologies, Chennai 600040, India (e-mail: swarna.ravindrababu@coovum.com).

D. C. Sekaran was with the Samsung R&D Institute India-Bangalore Pvt. Ltd., Bangalore 560037, India. He is now with Symbol Technologies India Pvt. Ltd., Bangalore 560103, India (e-mail: dhineshjim2006@gmail.com).

Digital Object Identifier 10.1109/JSEN.2016.2635691

step detection. Step detection in the GP developed by Jayalath *et al.* [18] uses a zero crossing method [18]. A step detection method based on pattern recognition, implemented using a finite state machine is developed by Lim *et al.* [19].

Such step detection methods based on accelerometer and gyroscope sensors are more common in wearable devices and smartphones. Most of these step detection methods for wearable devices require the sensing device to be mounted on to a specific location on the human body like user's feet [19]–[21], waist [4], [5], head [11], or arms, in order to accurately estimate the step count. However such a restriction on the placement of the device is not plausible for smartphones, as the user is free to handle the device in any of the modes.

While walking, a user can carry the smartphone in different modes, such as, holding in hand without swinging (henceforth referred as Texting mode), or holding in hand with swinging (henceforth referred as Hand Swinging mode), or placing it in either Shirt or Pant pocket (henceforth referred to as Shirt Pocket mode and Pant Pocket mode respectively), or mounting it on the waist. Among which the Texting, Shirt Pocket, Pant Pocket and Hand Swinging modes are the most common ones (henceforth collectively referred to as Smartphone usage modes). In each of these modes the smartphone can also be handled in different orientations. Existing accelerometer and gyroscope based pedometers for smartphones target at specific device usage modes and their performance is evaluated for these supported modes only. Some of these implementations also place a restriction on the orientation of the smartphone in each of these supported modes. In the works by Oshin and Poslad [14] and Alzantot and Youssef [15], an orientation independent AP specific to the Texting, Shirt Pocket and Pant Pocket modes is developed [14], [15]. On the other hand the AP developed by Bylemans *et al.* [16] and Kakiuchi and Kamijo [17] targets only the Pant Pocket mode of phone usage [16], [17]. GP developed by Jayalath *et al.* [18] targets slower walking conditions for pant pocket mode of phone usage, and is orientation dependent, where it is also necessary to keep a track of the gyroscope sensor axis which measures the angular movement of legs during walking [18].

Most of the pedometers available today do not address the necessity to work across different modes of smartphone usage and fail to provide a consistent accuracy across these modes. Specifically the Hand Swinging mode of phone usage is not taken into consideration in the development and evaluation. These shortcomings of the existing smartphone pedometers make it necessary to develop a pedometer solution that not only works accurately in all the modes of usage, but is also independent to the orientation in each of these modes. Based on the analysis of the accelerometer and gyroscope sensor data for different smartphone usage modes it has been observed that accelerometer sensor data provides accurate step detection in Texting, Shirt Pocket and Pant Pocket modes, while the gyroscope sensor data can be used for step detection in Pant Pocket and Hand Swinging modes. The presence of acceleration components from swinging motion of hands in the Hand Swinging mode cancels/distorts the patterns corresponding to true steps in the accelerometer sensor data, thereby causing the AP to count a reduced or incorrect number of steps. In Texting and Shirt Pocket modes the phone is placed at body locations that are not subjected to major angular movements during walking, thereby causing the angular magnitude sensed by the gyroscope sensor in accordance with the steps taken to be of lesser magnitude, which in-turn fails to assist in successful detection of steps. Hence by the integration of AP and GP, a solution that accurately detects steps across all the smartphone usage modes can be developed.

Random motions done by the user during phone handling also generate patterns in the sensor data that are similar to the ones generated due to steps. Since pedometers rely on counting these patterns, random motions will also be counted as valid steps, thereby providing an incorrect estimate of step count. Hence methods to discard these random motions also have to be considered in the development of pedometers.

In this paper we present UMOISP - Usage Mode and Orientation Invariant Smartphone Pedometer, a solution which provides an accurate estimate of step count in all the different modes and orientation a smartphone can be handled by the user. Specifically, Texting, Shirt Pocket, Pant Pocket and Hand Swinging modes will be considered. In UMOISP, orientation independent accelerometer and gyroscope pedometers are developed using pattern recognition methods. Further these two pedometers are integrated in a complementary fashion based on the device usage mode for consistent step detection across different modes. The proposed system also includes a magnetometer sensor based random motion detection algorithm that eliminates the false steps arising from the random motions generated by the user.

The results of the proposed UMOISP show accurate step counts in all the smartphone usage modes when tested across various walking conditions with different users. In walking conditions, UMOISP detects and counts steps with an average error of 0.48%, 1.16%, 1.98% and 1.46% in Texting, Shirt Pocket, Pant Pocket and Hand Swinging modes respectively. False step counts due to random motions from device handling are also mitigated by about 72%.

II. UMOISP SYSTEM

The robust pedometer implementation presented in this paper utilizes the step count information from the AP and GP in a complimentary fashion based on the smartphone usage mode. Further, a magnetometer based random motion detection algorithm integrated to this system improves the overall step count accuracy by eliminating the false steps arising from random phone handling. In the sections to follow, a detailed description of the proposed system will be presented. We begin by discussing the step detection methods used in the implementation of AP and GP. Later we present the architecture details of the UMOISP system. Finally the phone mode identification and random motion detection algorithms will be presented.

A. Accelerometer Pedometer - Step Detection

AP in the proposed system detects steps using pattern recognition method; and the implementation details of the same is

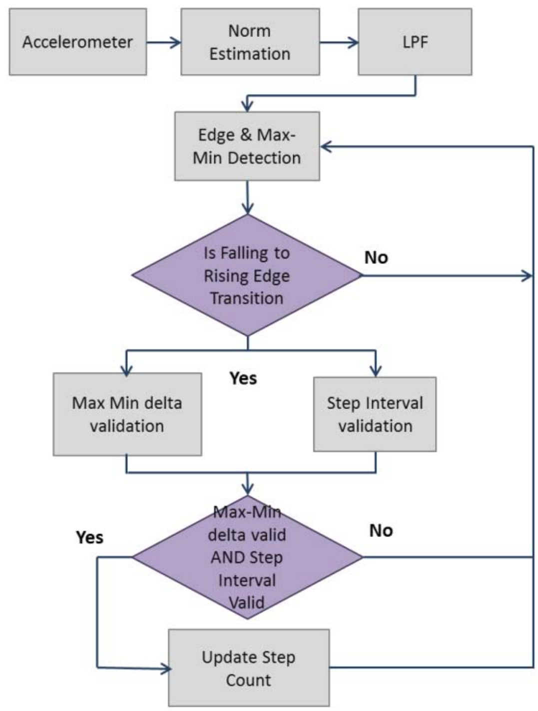

Fig. 1. Accelerometer Pedometer.

shown in Fig. 1. Accelerometer data along the 3 axes, sampled at 60 ms intervals, is combined together to estimate the norm of the acceleration values; this value is further compensated to remove the gravity component g_n (1). This component g_n is equal to the standard gravity value of 9.80665 m/s^2, which is the nominal gravitational acceleration of an object near the surface of the Earth. In the proposed approach relative magnitudes in the accelerometer data will be used for step detection rather than the absolute values. Hence the compensation of gravity component from the acceleration norm just shifts the mean value of the norm signal closer to zero. Usage of this norm signal for step detection, rather than the individual axis data, results in a pedometer implementation that is independent to the orientation of the smartphone. The computed norm signal will further be passed through a low pass filter to remove the high frequency noise components and to make the norm signal smoother so that the software implementation for counting the patterns become easier. The low pass filter is realized using a fifth order Butterworth filter with a cut-off frequency of 3 Hz. The choice of this cut-off frequency depends on the maximum value of step frequency that the pedometer must support. From the analysis of the experimental data collected with different users, average step frequencies of 1.5 Hz and 2.5 Hz were observed during walking and running conditions respectively, hence a cut-off frequency of 3 Hz, which caters to both these conditions, was chosen. The filtered signal (henceforth referred to as Post-processed signal) will further be used for step detection.

$$A_{NORM} = \sqrt{(A_x^2 + A_y^2 + A_z^2)} - g_n \tag{1}$$

where A_{NORM} is the acceleration norm; g_n is the gravity constant; A_x, A_y and A_z are the accelerometer values along the 3 axes.

In the proposed AP implementation step occurrences are detected by keeping track of the rising and falling edges along with the current local maxima and local minima. Fig. 2 illustrates the method of step detection on a sample Post-processed signal collected in Texting mode. Also the definitions of various terms associated with step detection are illustrated in Fig. 2. A step is said to have occurred whenever there is a transition from falling edge to rising edge provided there was a valid previous rising edge. The validity of the step is further verified by two checks, namely, a magnitude level check on the Post-processed signal, and a time duration check for step interval validity. The magnitude level check verifies whether at the instant of step occurrence the swing between the current local maxima and current local minima in the Post-processed signal exceeds a predefined threshold. This check eliminates the counting of small amplitude cycles that correspond to false steps as valid ones. The time duration check verifies whether the time interval between the step occurrences is within the practical limits of human motion dynamics. This check eliminates the counting of false steps that result in momentary or prolonged patterns of significant amplitude in the Post-processed signal as valid steps. The maximum and minimum threshold values of the step intervals used for the time duration check are 2 and 0.25 seconds respectively. Along with these fixed threshold values for step intervals, average step interval estimated over a predefined number of steps is also used to perform an additional time duration check. This time duration check is introduced after the user completes four successive steps since the start of the walk. Average step interval over the last four steps is computed at each step occurrence, which is further used to derive another set of run-time threshold values for time duration check. These run-time maximum and minimum values are chosen to be +/- 30% of the average step interval, which caters to varying step frequencies within a single stretch of walk. Only if the time duration between the steps satisfies both the fixed and run-time threshold value checks, the step is said to be valid. In addition to these a Step Sequence check is also added to improve false step rejection at the start of every walk. The condition for Step Sequence check is satisfied once the user takes four consecutive valid steps since the start of a walk. Only after the conditions for Step Sequence check are met the successive steps will be treated as valid and will be counted instantaneously as and when they occur. The four steps which are detected during the Step sequence check will also be included in the final step count.

B. Gyroscope Pedometer - Step Detection

The step detection method of the GP is similar to the one presented in the AP section. However one major difference is the way in which the Post-processed signal is derived. Samples from the gyroscope sensor along all the 3 axes, collected at intervals of 60 ms, are combined to obtain the norm of the angular rate (2). Similar to the AP, usage of the derived angular rate norm instead of individual axis data makes the step detection method orientation independent. The norm of the gyroscope angular rate will then be low pass filtered to

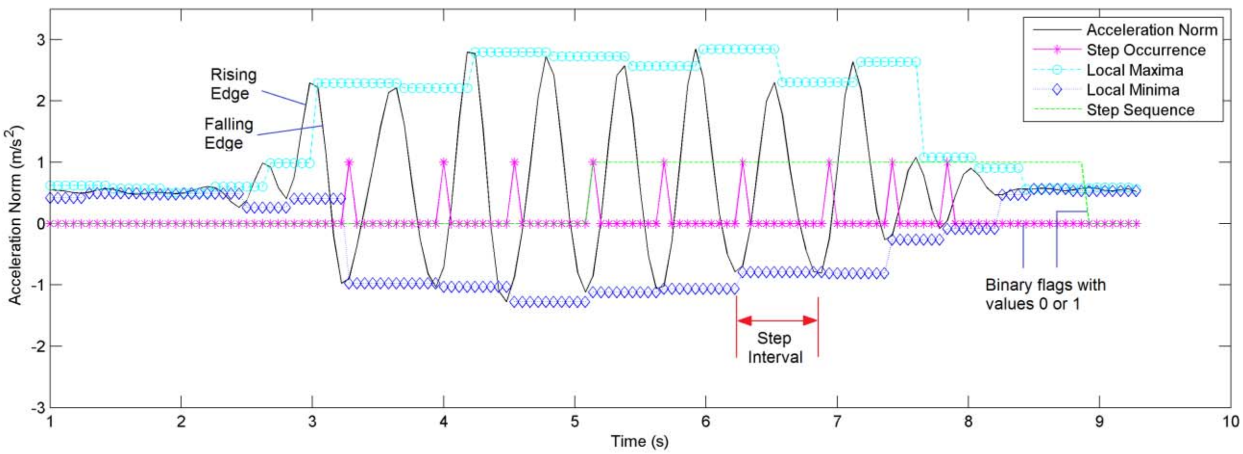

Fig. 2. Step detection on the Post-processed signal.

remove the high frequency components. The specifics of the low pass filter are the same as used in AP. This filtered signal will then be used for step detection. The subsequent procedure for step detection is the same as discussed in the AP implementation, except that the thresholds for magnitude level check in the Post-processed signal are based on angular rates rather than the acceleration values.

$$G_{NORM} = \sqrt{G_x^2 + G_y^2 + G_z^2} \tag{2}$$

where G_{NORM} is the angular rate norm; G_x, G_y and G_z are the gyroscope sensor values along the 3 axes.

C. UMOISP System Architecture

The step detection performance of AP and GP in different phone usage modes is shown in Fig. 3 and Fig. 4 respectively. It can be observed that the AP works well in the Texting, Shirt Pocket and Pant Pocket modes; while the GP works better in Pant Pocket and Hand Swinging modes. In the sections to follow, analysis of the performance of AP in different modes will be discussed first, followed by the analysis of GP, finally the motive for integration of these two pedometers and the architecture details of UMOISP will be presented.

While walking, across each step, human body is subjected to acceleration and de-acceleration forces, which are along the forward, lateral and vertical directions, as shown in Fig. 5. Among these, the forces in vertical and forward directions are more prominent. These forces, observed during the stance and swing phases of a step, generate patterns in the forward and vertical directions, which are measured by the accelerometer sensor. It is the patterns along these directions, which significantly contribute towards the cyclic patterns observed with steps in the Post processed signal, thereby playing a major role in determining the step detection accuracy. In order to study the variation of these patterns with different phone usage modes, and also evaluate their impact on step detection, we perform an analysis of the accelerometer sensor data and the derived norm signal for each of the usage mode. For the analysis, accelerometer sensor data was collected in all the smartphone usage modes, namely, Texting, Shirt Pocket, Pant Pocket and Hand Swinging modes, and the axes conventions for the sensor data (both accelerometer and gyroscope) is as shown in Fig 5.

In Texting, Shirt Pocket and Pant Pocket modes the accelerometer sensor measures only the forces that are due to the dynamics of human steps. Depending on the orientation of the device, one, or two, or all the three axes of the accelerometer sensor can measure the acceleration experienced in forward and vertical directions. This is illustrated by the accelerometer sensor data in Fig. 6a, Fig. 6b and Fig. 6c, which correspond to the data collected in Texting, Shirt Pocket and Pant Pocket modes respectively. The accelerometer sensor data along the individual axes, as illustrated in these figures, is the low pass filtered version of the raw sensor data, obtained using the filter specifications mentioned in the Accelerometer Pedometer-Step Detection section. The data for Texting mode was collected by holding the phone flat on hand, such that the individual axes of the accelerometer, namely x, y and z, correspond to lateral, forward and vertical directions respectively. In Shirt Pocket mode, the phone was placed vertically in shirt pocket such that the accelerometer axes x, y and z correspond to lateral, vertical and forward directions respectively. In Pant Pocket mode, the phone was placed inside the pant pocket such that when the user is stationary, the accelerometer axes x, y, and z correspond to lateral, vertical and forward directions respectively. In Texting and Shirt Pocket modes the orientation of the phone is constant within a step, and the accelerometer norm signal (Post-processed signal) used for step detection as illustrated in Fig. 6a and Fig. 6b, has significant contribution from components along vertical and forward directions, with vertical direction component being more prominent. Although in Pant Pocket mode, the orientation of the phone changes within a step, the norm signal contains significant contributions from the components in vertical and forward directions only. This is illustrated in Fig.6c, where the combination of forward and vertical components is sensed by both the accelerometer axes, namely z and y. In each of these modes, as illustrated in Fig. 6a, Fig. 6b and Fig. 6c, the norm signal obtained from the accelerometer data during walking is free from any distortions and consists of patterns that correspond to steps only, making the step detection method reliable and accurate.

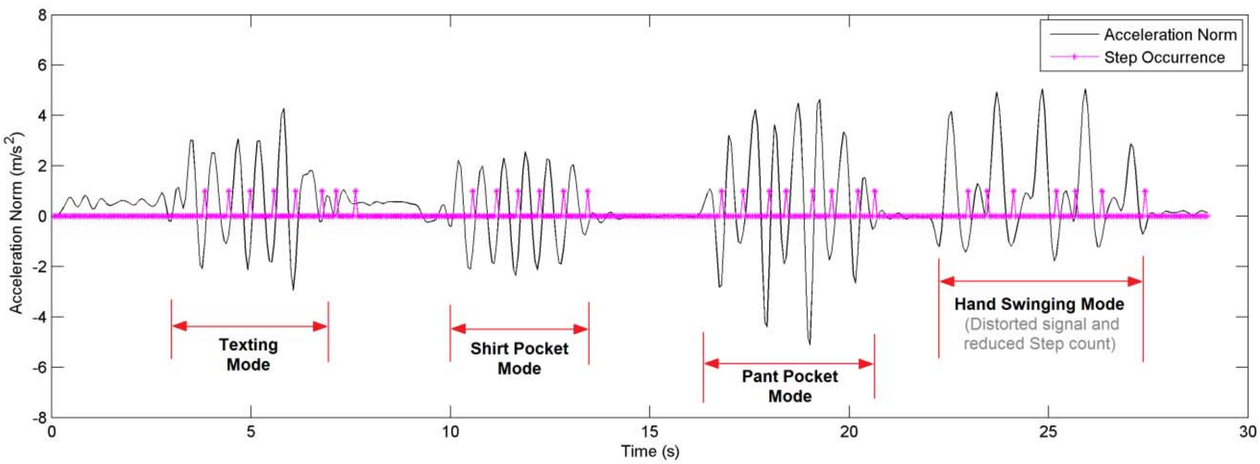

Fig. 3. Step detection by Accelerometer Pedometer in different modes.

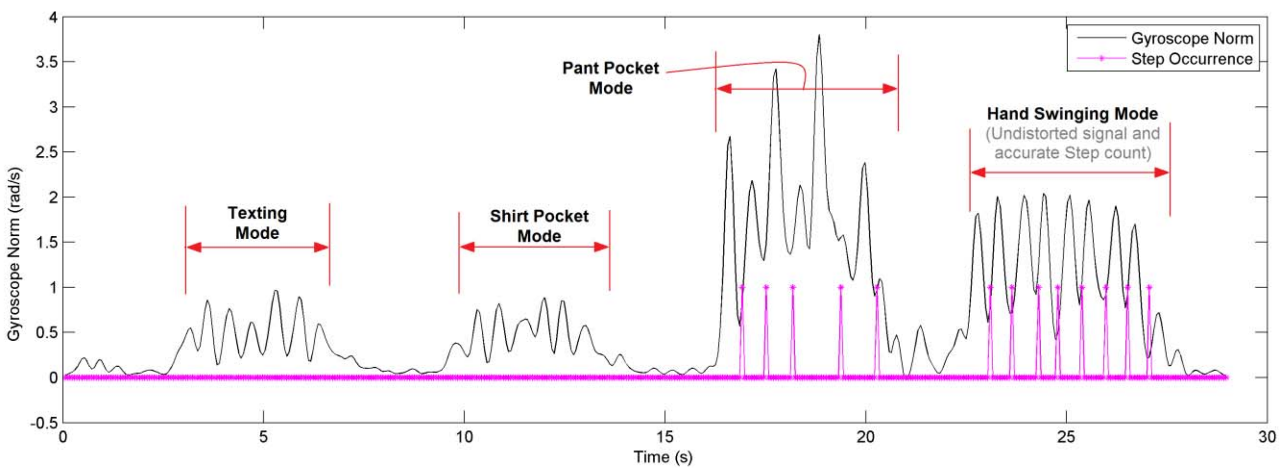

Fig. 4. Step detection by Gyroscope Pedometer in different modes.

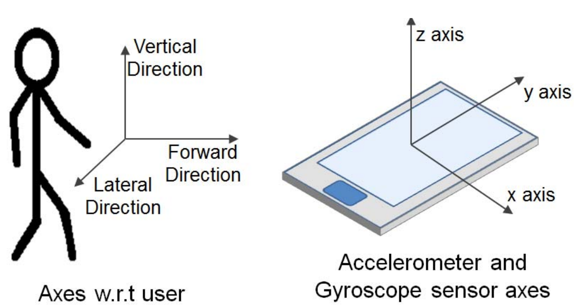

Fig. 5. Definition of axes.

In Hand Swinging mode, the accelerometer sensor measures not only the forces due to dynamics of human steps, but also the components that arise due to swinging motion of hands. These components from hand swinging motion, though synchronized to the step occurrence instants, distort the patterns corresponding to steps in the acceleration data along the vertical and forward directions. Fig. 6d illustrates the effect of hand swinging motion on the Post-processed signal. This data was collected by holding the phone in Hand Swinging mode, such that, when the user is stationary the accelerometer axes x, y and z are aligned to the vertical, forward and lateral directions respectively. As shown in Fig. 6d, the patterns corresponding to steps are distorted at many of the step occurrence instants by the hand swinging motion. Step detection algorithms can be modified to count these distorted patterns in the norm signal as steps; however, such algorithms fail to provide consistent results and also perform poorly in false step rejection.

On the other hand the GP provides accurate step detection in Pant Pocket and Hand Swinging modes, but fails in Texting and Shirt Pocket modes. In Pant Pocket and Hand Swinging modes the phone is carried by the users such that the gyroscope sensor can directly measure the movements of body parts that are subjected to significant angular motion during walking, namely the legs taking steps and the hands which are swinging. These movements are synchronized with the steps taken and are picked up effectively by the gyroscope sensor present in the phone, facilitating an accurate detection of steps. However in Texting and Shirt Pocket modes the magnitude of angular rates in the patterns generated by the gyroscope in accordance with the steps taken will be of lesser magnitude, and are not consistent with the number of steps taken. A step detection method designed to count these patterns of smaller magnitudes also performs poorly in false step rejection, as the patterns resulting

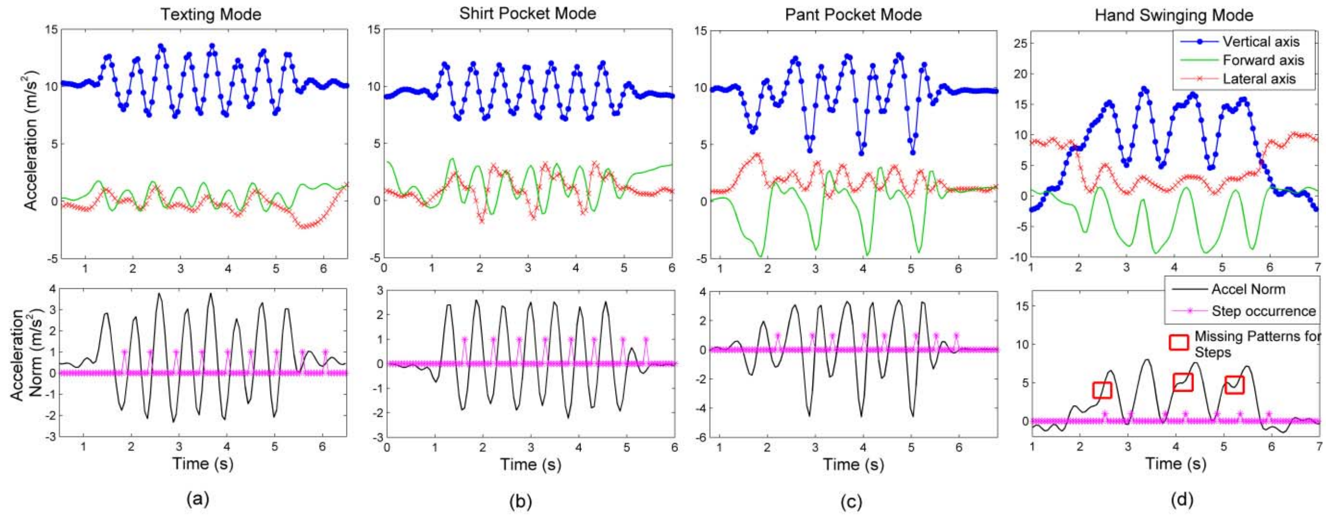

Fig. 6. Accelerometer data in various phone usage modes while walking on flat surfaces: (a) Texting mode, (b) Shirt Pocket mode, (c) Pant Pocket mode and (d) Hand Swinging mode.

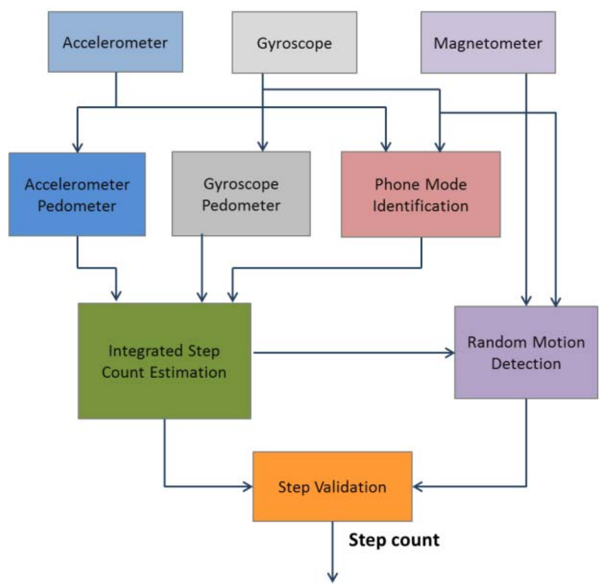

Fig. 7. UMOISP architecture.

from phone handling also have similar characteristics. Hence by the integration of orientation independent AP and GP, in a complementary fashion based on the phone usage mode, a robust pedometer which is consistent and accurate in all the phone usage modes can be developed.

The architecture diagram of the proposed system, UMOISP, is shown in Fig. 7. In the developed system, step detection in Texting, Shirt Pocket and Pant Pocket modes is handled by the AP, while the GP is used for Hand Swinging mode. A phone mode identification block determines whether the phone carried by the user when walking is in Hand Swinging or any other mode (Texting, Shirt Pocket or Pant Pocket mode). The steps detected by the AP and GP are then utilized in a complimentary fashion by the step count integration block to obtain the integrated(or final) step count based on the inputs from the phone mode identification block. Detailed description of the step detection methods used in AP and GP are mentioned in sub-sections A and B of Section II. The step count integration block allows the output from either GP or AP depending on whether the phone is in Hand Swinging or any other modes respectively. The step count integration block maintains an estimate of the integrated step count which is updated in real-time with step occurrences. This block also handles the task of accurately maintaining the integrated step count in situations where the user changes the phone mode during walking. Any false steps counted due to random motions will further be corrected by the Step validation block based on the results of the random motion detection algorithm.

D. Random Motion Detection

Random motions done by the user when stationary also generate patterns similar to valid steps in the accelerometer and gyroscope sensor data. In the proposed pedometer implementation a random motion detection algorithm is developed to detect and eliminate such false steps caused due to random phone handling by the user. This algorithm is based on the fact that, whenever a user is walking in an indoor or outdoor environment, across successive steps, the overall magnetic field present around the user will vary by a significant amount. In indoors, presence of re-enforced concrete and steel structures, permanent magnetic and ferromagnetic materials, and, the magnetic fields arising from manmade electrical appliances, will cause huge variations in the local magnetic field [22], [23]. On the other hand in outdoor conditions these variations will be relatively smaller than indoors, since the otherwise constant ambient earth's magnetic field will have smaller perturbations due to changes in the deposits of earth's crust and the presence of sporadic man made ferrous structures [22].

However, when the user is stationary and is doing some random motions the overall magnetic field variations across

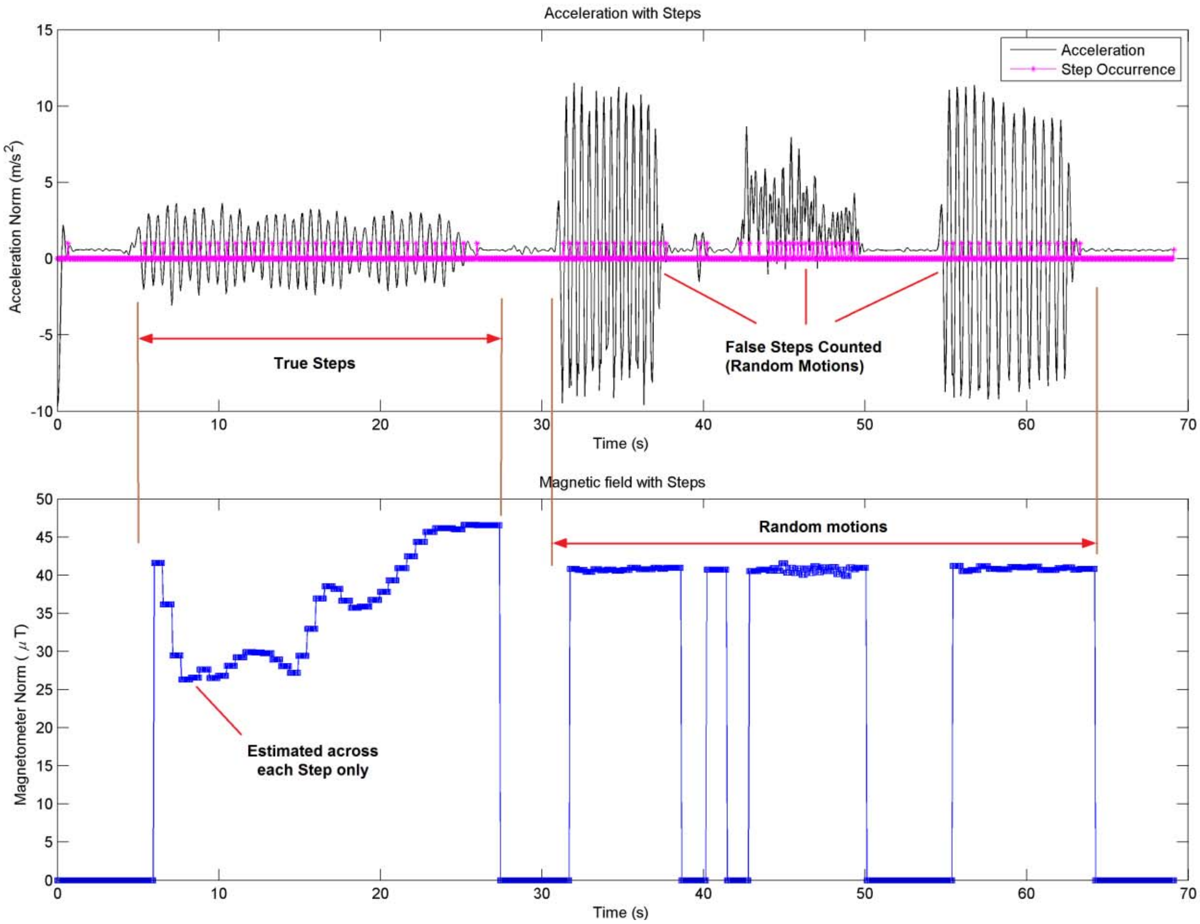

Fig. 8. Magnetic Field variation across Steps and Random motions.

each false steps being detected will be of very less magnitude, which is much lower than what is observed while walking in outdoor conditions. A sample plot that depicts these magnetic field variations during walking and stationary conditions (collected indoors on Samsung Galaxy S5 device) is illustrated in Fig. 8. Hence, by choosing a suitable threshold for these magnetic field variations, false steps from random motions can be easily identified. In the UMOISP, steps will be counted as valid only when the magnetic field variations across a predefined number of previous successive steps cross a certain threshold, otherwise they will be ignored as false steps due to random motions.

Implementation of the random motion detection algorithm in UMOISP is illustrated in Fig. 9. Magnetometer data along the 3 axes, sampled at 60ms intervals, is combined together to estimate the norm of magnetic values, M_{NORM} (3). Further, based on the step occurrence information from the Integrated Step count block, these values (M_{NORM}) are averaged over each step interval to compute the average magnetic norm $m(k)$ (4). Such average norm values computed across the current ($m(n)$) and each of the last 3 consecutive steps (namely $m(n-1)$, $m(n-2)$ and $m(n-3)$) are used to estimate the magnetic field variation M (5). If this magnetic field variation exceeds the predefined threshold, $M_{TrueSteps}$, the current step is said to be a valid one and will be counted or else it will be discarded as a false step due to random motion. A value of 1.2 μT is used for the threshold $M_{TrueSteps}$, which was chosen based on the analysis of the magnetometer data collected in various indoor and outdoor test conditions.

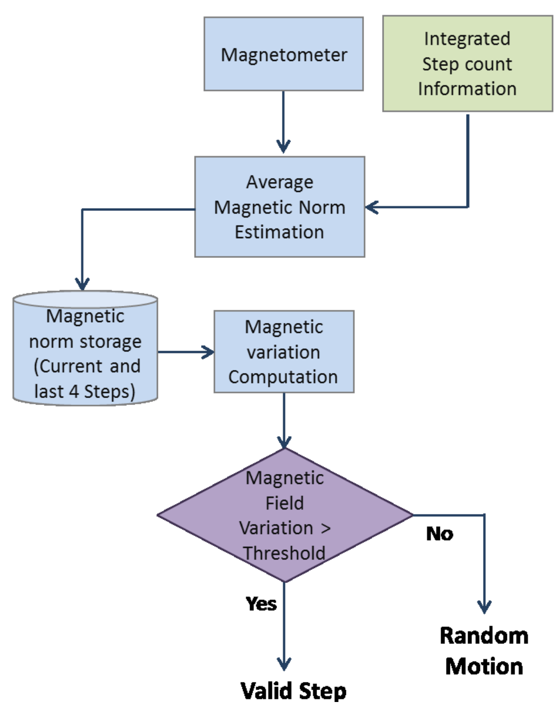

Fig. 9. Random Motion detection algorithm.

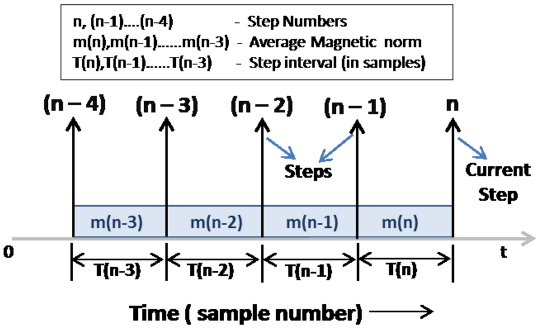

Fig. 10. Timing diagram to illustrate the Computation of Magnetic field variation across steps.

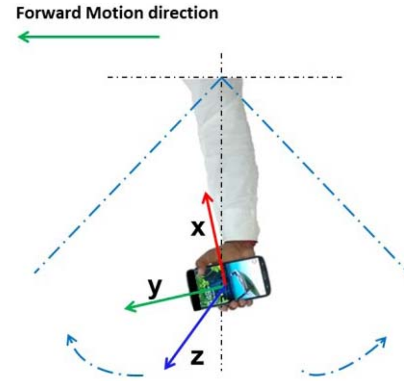

Fig. 11. Phone usage in Hand Swinging mode and axes convention.

The definitions of various terms associated with random motion detection are better illustrated with the Timing diagram in Fig. 10.

$$M_{NORM} = \sqrt{(M_x^2 + M_y^2 + M_z^2)} \tag{3}$$

where M_{NORM} is the magnetic field norm;M_x, M_y and M_z are the magnetometer sensor values along the 3 axes.

$$m\,(k) = \frac{\sum_{T(k)} M_{NORM}}{T\,(k)} \tag{4}$$

where $m\,(k)$ is the average of magnetic norm M_{NORM} (3) in μT, estimated over the samples in step interval $T\,(k)$, between the k^{th} step and the $(k\text{-}1)^{\text{th}}$ step (i.e., previous step).

$$M = \sqrt{\sum_{k=0}^{2} \left(m\,(n-k) - m\,(n-(k+1))\right)^2} \tag{5}$$

where M is the Total magnetic field variation in μT

E. Phone Mode Identification

A phone mode identification algorithm that identifies the Hand Swinging mode of phone usage is developed to assist the integration of the Accelerometer and Gyroscope Pedometers. Data collected from both the accelerometer and gyroscope sensors in Texting, Shirt Pocket, Pant Pocket and Hand Swinging modes were analyzed to identify the features that can be used for mode classification. The selected features were the mean values of accelerometer sensor data and the peak to peak swing of the gyroscope sensor data along all the three sensor axes observed over a time duration of one second. The choice of these set of features was motivated by the work from Bhatt *et al.*, using which the authors were able to develop a context classification system with an accuracy of 92% [24]. As opposed to the SVM based approach [24] for context classification, we chose to use a threshold based approach, since the number of modes to be classified is only two, namely the presence or absence of Hand Swinging mode. Whenever a user carries the phone in Hand Swinging mode, the swinging motion of the user's hand will be sensed by the gyroscope axis which is normal to the plane containing the periodic swinging motion of the hand. For the smartphone with axes conventions as shown in Fig. 11, these swinging motions will be picked up by the gyroscope along the z axis. In the proposed implementation of the threshold based approach for Hand Swinging mode identification, the gyroscope values along the z axis are monitored over a duration of one second to estimate the magnitude of peak to peak swing in the angular rate by taking a difference of local maxima and local minima. Whenever the magnitude of this swing value crosses a predefined threshold the phone is said to be in Hand Swinging mode. This predefined threshold value was chosen from the analysis of the gyroscope sensor data collected in Hand Swinging mode (on a Samsung Galaxy S5 device) with different male and female users of varying height and age. As illustrated in Table I, the threshold based Hand Swinging mode identification algorithm is able to achieve a classification accuracy of 94.62%.

TABLE I
CONFUSION MATRIX FOR THE HAND SWINGING MODE CLASSIFICATION

Total Samples : 11790		Predicted Values		
		Hand Swinging mode	Other modes	
Actual Values	Hand Swinging mode	4769	271	5040
	Other modes	180	6570	6750
		4949	6841	

III. TESTS AND EXPERIMENTAL RESULTS

In order to evaluate the performance of the proposed algorithm an android application for the UMOSIP was developed on a Samsung Galaxy S5 smartphone. This device has 3 axes accelerometer, gyroscope and magnetometer sensors. The accelerometer and Gyroscope sensors used in this phone are from InvenSense (MPU 6500), while the magnetometer is from Asahi Kasei Microdevices (AK09911C). The android implementation for the UMOISP registers for all the three sensors at a sampling rate of 60 ms. At each of these sample epochs, the UMOISP step detection algorithms (as discussed in Section II) are run to provide a real time estimate of the step count. Accuracy of the UMOISP was tested in both walking and running conditions. In the tests for walking conditions, Texting, Shirt Pocket, Pant Pocket and Hand Swinging modes were considered, while for running only Pant Pocket

TABLE II

UMOISP WALKING TEST RESULTS – MALE PARTICIPANTS

User	Age (years)	Height (m)	Texting mode				Shirt Pocket mode				Pant Pocket mode				Hand Swinging mode			
			Flat Surface		On Stairs		Flat Surface		On Stairs		Flat Surface		On Stairs		Flat Surface		On Stairs	
			TSC[a]	USC[b]	TSC	USC	TSC	USC	TSC	USC	TSC	USC	TSC	USC	TSC	USC	TSC	USC
1	28	1.77	188	186	100	101	202	202	105	104	179	180	100	99	182	182	100	101
2	27	1.65	194	194	100	95	185	185	100	91	188	189	103	99	200	201	102	98
3	29	1.82	188	188	100	101	185	186	138	142	192	194	108	110	215	210	100	97
4	29	1.71	190	190	107	107	188	188	113	111	189	191	112	112	197	197	109	109
5	24	1.80	200	200	129	129	202	202	130	131	197	200	129	131	200	198	130	131
6	25	1.62	198	198	144	139	192	197	142	144	193	197	132	137	193	191	133	135
7	27	1.80	186	184	128	129	184	187	127	129	184	184	129	130	185	189	131	132
8	28	1.82	197	196	133	134	196	197	134	134	198	198	136	136	200	202	133	129
9	29	1.78	201	201	138	138	204	204	140	146	201	205	142	142	197	199	139	143
10	29	1.65	219	219	132	132	208	209	124	126	206	208	126	125	202	197	130	129
11	30	1.65	187	187	135	135	183	183	135	136	184	184	135	138	179	180	142	136
12	31	1.83	187	186	137	136	182	185	136	136	185	185	136	138	186	189	136	140
13	31	1.72	196	196	131	131	190	190	130	131	188	192	131	134	186	188	133	133
14	32	1.62	216	216	145	145	210	213	137	141	211	214	147	152	218	220	142	145

TSC[a] refers to True Step Count, USC[b] refers to UMOISP Step count

TABLE III

UMOISP WALKING TEST RESULTS – FEMALE PARTICIPANTS

User	Age (years)	Height (m)	Texting mode				Pant Pocket mode				Hand Swinging mode			
			Flat Surface		On Stairs		Flat Surface		On Stairs		Flat Surface		On Stairs	
			TSC	USC	TSC	USC	TSC	USC	TSC	USC	TSC	USC	TSC	USC
1	23	1.60	247	249	129	130	247	255	132	136	242	242	132	135
2	23	1.52	215	218	127	129	216	226	128	132	220	220	129	129
3	24	1.57	214	215	129	130	212	215	134	140	208	210	135	142
4	28	1.57	210	211	132	132	219	223	132	134	200	201	134	122
5	29	1.69	221	221	136	136	216	222	137	141	217	225	135	135

mode was considered. In each of these tests, results from the UMOISP were compared with the step counts obtained from the S Health [1] and Google Fit (G Fit) [2] applications, which were also installed on the same device used for testing

In order to evaluate the performance of the system under varying gait characteristics, male and female participants, with different age and height were used to perform the walking tests. The number of male participants used for the testing was 14, whose age and height were in the range of 24 to 32 years and 1.62 m to 1.83 m respectively. On the other hand, five female users participated in the tests, whose age and height were in the range 23 to 29 years and 1.52 m to 1.69 m respectively.

With each of the male participants, the performance of UMOISP was tested in real time for Texting, Shirt Pocket, Pant Pocket and Hand Swinging modes. While, for the female participants, these tests were carried out in Texting, Pant Pocket and Hand Swinging modes only, as these correspond to the most commonly used modes by female users, among all the modes supported by UMOISP. These walking tests were performed in different walking surface conditions commonly found indoors, namely flat surfaces and staircases, to test the adaptability of the UMOISP. In each of these tests the users were asked to hold the phone in a certain mode and walk along a predefined trajectory. To complete the test trajectory chosen on a flat surface, users on an average took around 195 steps. The trajectory chosen for tests on stairs was such that, the users had to walk two floors in both upward and downward directions, where they took on an average around 125 steps to complete the trajectory. During each of these tests, the true number of steps taken by users was also counted. At the end of each test, step counts obtained from the UMOISP, S Health and Google Fit applications were noted.

The performance of UMOISP in the walking tests with different male and female users is listed in Table II and Table III respectively. Comparison of the step count accuracies (expressed as the ratio of step count estimated by the application to the true value) obtained with the UMOISP, S Health [1] and Google Fit [2] applications, for different male and female users, across different modes is illustrated in Fig. 12 and Fig. 13 respectively. It can be observed from the test results that the accuracy of UMOISP is better for most of the users. However this accuracy improvement is more apparent in Hand Swinging mode for users who swing their hands by a significant amount when walking, which can be observed in the tests results for few male (User 6, User 8 and User 13) and female (User 4) users as illustrated in Fig. 12 and Fig. 13 respectively.

The inclusion of the GP for step counting in Hand Swinging mode improved the overall accuracy of UMOISP. Integration of AP and GP in the UMOISP, to provide a consistent step count accuracy across all smartphone usage modes is also illustrated in Fig. 14 which compares the step count accuracies of UMOISP, AP and GP in different phone usage modes for male users in flat surface walking conditions.

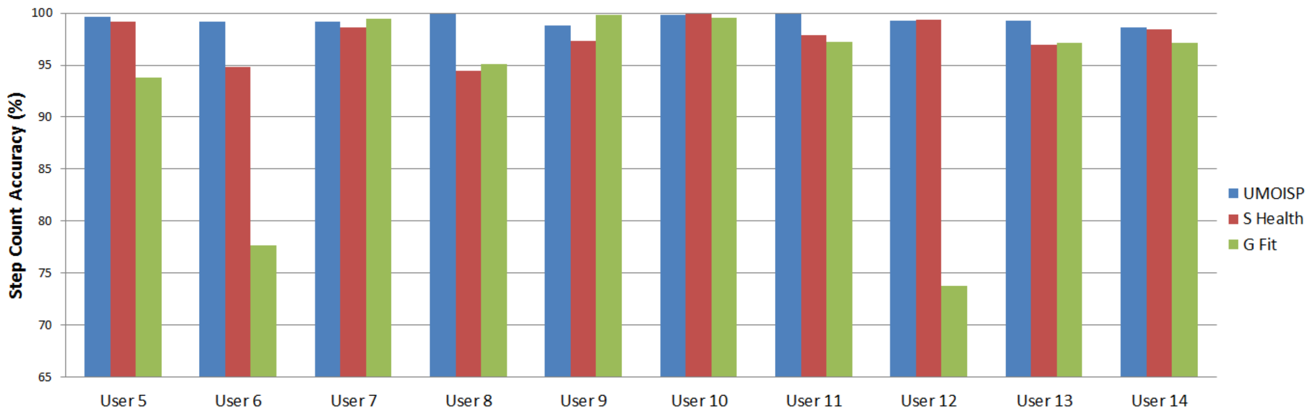

Fig. 12. Step Count Accuracy comparison for Male users in walking conditions (on flat surfaces and stairs).

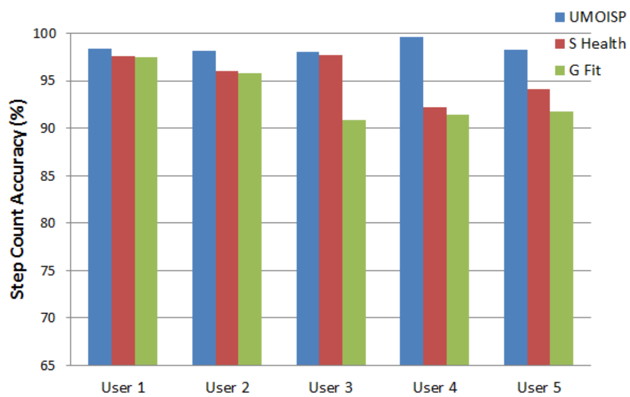

Fig. 13. Step Count Accuracy comparison for Female users in walking conditions (on flat surfaces and stairs).

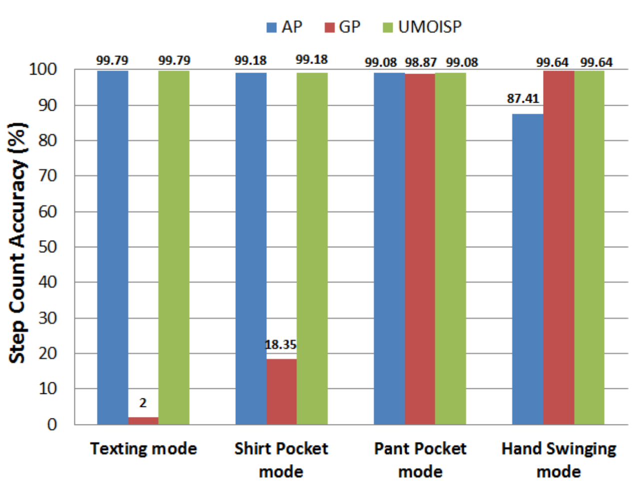

Fig. 14. UMOISP, AP and GP step count accuracy comparison.

A significant observation made during the tests performed on stairs with the Hand Swinging mode of device usage was that, the users do not swing their hand with the same magnitude on stairs as they would while walking on flat surfaces. In-fact the magnitude of Hand Swinging motion on stairs is much lesser than the ones observed during walking on flat surfaces. This was also clearly evident from the gyroscope sensor data observed during these tests. As a result, in Hand Swinging mode, the GP works with a reduced accuracy on stairs when compared with the walks on flat surfaces. However the AP was able to accurately pick up all the foot impacts on stairs, leading to reliable step detection even in Hand Swinging mode. Hence in the evaluation of the UMOISP for tests on stairs, only AP was used across all the modes of device usage. There are several activity recognition methods, which can classify whether the user is walking on flat surfaces or on stairs, with reasonable accuracies, using accelerometer and gyroscope sensors only [25]–[27]. Any of such activity recognition methods can further be integrated with the UMOISP to enable the automatic usage of AP for Hand Swinging modes on stairs, thereby improving the overall step count accuracy.

Although Google Fit application showed a comparable accuracy on flat surfaces, it relied on post processing to detect steps and also took a while to update the step count, unlike the UMOISP and S Health applications which instantaneously updated the steps as and when they were taken. This makes Google Fit unsuitable for applications like Pedestrian navigation system which need a real time update of steps. During the tests on stairs, it was also observed that the Google fit application counted lower number of steps for few of the users in some of the modes, and these are evident in the results for the male (User 6, User 12) and female (User 3) users as illustrated in Fig. 12 and Fig. 13 respectively.

In the walking tests, UMOISP was able to count steps for male participants with an average error of 0.34%, 1.16%, 1.08% and 1.12% in Texting, Shirt Pocket, Pant Pocket and Hand Swinging modes respectively. On the other hand, for the female participants the observed error was 0.62%, 2.88% and 1.79% in Texting, Pant Pocket and Hand Swinging modes respectively. Average of these step count errors across male and female users, in different phone usage modes is listed in Table V. From Table V it can also be observed that the UMOISP accuracy is slightly degraded in Shirt Pocket, Pant Pocket and Hand Swinging modes when compared with the Texting mode. A similar behavior was also seen with the S Health and Google Fit applications. All these applications counted few additional steps along with the true steps in Shirt Pocket, Pant Pocket and Hand Swinging modes, there

TABLE IV
RANDOM MOTION DETECTION TEST RESULTS

User	Age (years)	Height (m)	True Step Count	UMOISP with Random motion detection		UMOISP without Random motion detection	
				Step count before performing random motions	Step count after performing random motions	Step count before performing random motions	Step count after performing random motions
1	32	1.72	203	204	218	204	257
2	23	1.75	205	209	222	209	254
3	29	1.65	202	208	222	208	263
4	28	1.65	203	204	223	204	269
5	27	1.67	200	200	217	200	260

TABLE V
UMOISP TEST RESULTS ACROSS DIFFERENT USERS AND TEST SCENARIOS

Testing Mode	UMOISP Step Count Average Error
Texting	0.48%
Shirt Pocket	1.16%
Pant Pocket	1.98%
Hand Swinging	1.46%

TABLE VI
UMOISP RUNNING TEST RESULTS

User	Age (years)	Height (m)	Pant Pocket mode	
			TSC	USC
1	25	1.62	315	312
2	27	1.80	244	249
3	30	1.65	298	299
4	31	1.72	320	322
5	32	1.62	315	319

by leading to a slightly degraded accuracy. Patterns generated in the sensor data, while placing the phone in to the pocket, as well as while moving out the phone from pocket, led to additional step counts in Shirt Pocket and Pant Pocket mode tests. Likewise, in the Hand Swinging mode, the patterns generated during the transition of phone to the swinging hand, at the start and end of each test, resulted in additional steps. Overall in these walking tests, UMOISP was able to count steps with an average error of 1.07% across all the participants (male and female), while the errors for S Health and Google Fit were 3.38% and 6.74% respectively.

Tests were also done on a treadmill to evaluate the UMOISP performance in running conditions. Usually, while running, people prefer to carry their phones either in pant pocket or with a band which is strapped on to their hands. Since Pant Pocket mode is one of the modes supported by UMOISP, the running tests were performed in this mode only. In each of these tests which lasted for about 3 minutes, the users started off from a stationary state, later the speed of the tread mill was gradually increased so that the users started to walk. With further increase in the treadmill speed the users started running. After the users ran for around 1.5 minutes at an average speed of 8 km/hr, speed of the tread mill was gradually decreased so that the users slowly transitioned to the walking state, and then finally to the stationary state. During each of these tests the true steps taken by the user were also noted down along with the step counts from the UMOISP, S Health and Google Fit applications. Performance of UMOISP in running tests with different male users is listed in Table VI. Even in the running tests, UMOISP was able to count steps with a better accuracy when compared with the reference applications. UMOISP counted steps with an error of 1.05%, while the errors for S Health and Google Fit were 1.86% and 2.15% respectively.

Tests were also done to evaluate the performance of the random motion detection algorithm. The objective of these tests was not only limited to evaluate the accuracy of the algorithm in identifying false steps due to random motions, but also to verify the consistency of the algorithm in treating the true steps as valid ones across all the smartphone usage modes. Five male users participated in these tests, and in each of the tests, users were asked to walk around 200 steps, taking about 50 steps in each of the smartphone usage mode. The users started off the test by holding the smartphone in Hand mode and took around 50 steps, later after being stationary for a while placed the smartphone in Shirt Pocket mode and took around 50 more steps before coming back to stationary state again. The same procedure was further continued for Pant Pocket and Hand Swinging modes, thereby covering all the modes in a single test. Further, towards the end of each test, users were also asked to do some random motions holding the device in hand while being stationary. This completes the test trajectory for a single user, which was later repeated with 4 more additional users. During each of these tests the step counts were noted down at two instants, first when the user is stationary after completing the walk in Hand Swinging mode (i.e., after covering all the smartphone usage modes), and later towards the end of the test (i.e., after performing random motions).

The results of these tests, which compare the accuracy of UMOISP with and without random motion detection algorithm, are illustrated in Table IV. In counting the true steps, an average accuracy of 98.8% is obtained from both the variants of the UMOISP (with and without random motion detection algorithm). These results indicate that the inclusion of random motion detection algorithm to UMOISP does not have any impact on the normal counting of true steps across all the smartphone usage modes. Further, the test results also indicate that with the inclusion of the random motion detection algorithm, UMOISP can mitigate the false steps due to random motions to a greater extent, by around 72%.

The results from the above tests show that the proposed UMOISP detects steps with an accuracy of 98.73%, across all smartphone usage modes in walking conditions. Specifically the UMOISP was able to detect steps with an average error of 0.48%, 1.16%, 1.98% and 1.46% in Texting, Shirt Pocket, Pant Pocket and Hand Swinging modes respectively, when tested across different users in different walking conditions. In the running mode UMOISP achieved a step count accuracy of 98.95%. The test results also show that the UMOISP was able to discard around 72% percent of the random motions made by the users. Overall the developed algorithm for UMOISP was able to outperform the existing Pedometer applications. This robustness is achieved by combining the AP and GP, together with monitoring the phone usage mode.

IV. Conclusion

In this paper we presented UMOISP, a system that integrates Accelerometer and Gyroscope pedometers. Depending on the phone usage mode either of these two pedometers was used for step detection. On flat surfaces, Accelerometer Pedometer of the UMOISP provided accurate step counts in Texting, Shirt Pocket and Pant pocket modes, while the Gyroscope Pedometer worked accurately in Hand Swinging mode. However, on stairs, Accelerometer Pedometer was accurate across all the phone usage modes. With the integration of these two pedometers, UMOISP was able to achieve an overall step count accuracy of 98.73% in walking cases, when tested with various male and female users across different device usage modes in different walking surface conditions. Even in running conditions UMOISP achieved a step count accuracy of 98.95%. False steps from random motions were also discarded by about 72% with the inclusion of a magnetometer based algorithm.

Efficient handling of phone mode changes during walking, tighter integration (other than complementary fusion) of Accelerometer and Gyroscope Pedometers for improved accuracy, implementation and integration of walking surface identification algorithm and random motion algorithm tuning for efficient operation in environments with static magnetic fields (on a treadmill) would be the focus areas for future work.

Acknowledgment

The authors would like to thank the GPS/LBS team members at Samsung R&D Institute India – Bangalore for their support in the test data collection.

References

[1] Samsung Electronics Co., Ltd., Yeongtong-Gu, Suwon-Si, Gyeonggi-Do, Korea. (2015). *S Health, Version 4.1.1.0047 for Android*. [Online]. Available: https://play.google.com/store/apps/details?id=com.sec.android.app.shealth

[2] Google Inc., Mountain View, CA, USA. (2015). *Google Fit*. [Online]. Available: https://play.google.com/store/apps/details?id=com.google.android.apps.fitness

[3] F. Lin, A. Wang, C. Song, W. Xu, Z. Li, and Q. Li, "A comparative study of smart insole on real-world step count," in *Proc. IEEE Signal Process. Med. Biol. Symp. (SPMB)*, Philadelphia, PA, USA, Dec. 2015, pp. 1–6.

[4] C. G. Park, S. H. Shin, H. S. Hong, and J. G. Park, "Adaptive step length estimation with awareness of sensor equipped location for PNS," in *Proc. 20th Int. Tech. Meeting Satellite Division Inst. Navigat. (ION GNSS)*, Fort Worth, TX, USA, 2007, pp. 1845–1850.

[5] H. Leppäkoski, J. Collin, and J. Takala, "Pedestrian navigation based on inertial sensors, indoor map, and WLAN signals," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Kyoto, Japan, Mar. 2012, pp. 1569–1572.

[6] *Samsung Galaxy S5(SM-G900) User Manual*, Samsung Electronics Co., Ltd., Yeongtong-Gu, South Korea, 2014.

[7] J. Lester, C. Hartung, L. Pina, R. Libby, G. Borriello, and G. Duncan, "Validated caloric expenditure estimation using a single body-worn sensor," in *Proc. 11th Int. Conf. Ubiquitous Comput.*, Orlando, Florida, USA, Sep./Oct. 2009, pp. 225–234.

[8] R. W. Levi and T. Judd, "Dead reckoning navigational system using accelerometer to measure foot impacts," U.S. Patent 5 583 776, Dec. 10, 1996.

[9] R. Jirawimut, P. Ptasinski, V. Garaj, F. Cecelja, and W. Balachandran, "A method for dead reckoning parameter correction in pedestrian navigation system," *IEEE Trans. Instrum. Meas.*, vol. 52, no. 1, pp. 209–215, Feb. 2003.

[10] Q. Ladetto, "On foot navigation: Continuous step calibration using both complementary recursive prediction and adaptive Kalman filtering," in *Proc. 13th Int. Tech. Meeting Satellite Division Inst. Navigat. (ION GPS)*, Salt Lake City, UT, USA, 2000, pp. 1735–1740.

[11] S. Beauregard and H. Haas, "Pedestrian dead reckoning: A basis for personal positioning," in *Proc. 3rd Workshop Positioning, Navigat. Commun. (WPNC)*, Hannover, Germany, 2006, pp. 27–35.

[12] N. Zhao. (Jun. 2010). *Full-Featured Pedometer Design Realized With 3-Axis Digital Accelerometer. Analog Dialogue*. [Online]. Available: http://www.analog.com/library/analogdialogue/archives/44-06/pedometer.html

[13] F. Li, C. Zhao, G. Ding, J. Gong, C. Liu, and F. Zhao, "A reliable and accurate indoor localization method using phone inertial sensors," in *Proc. 14th Int. Conf. Ubiquitous Comput. (Ubicomp)*, Pittsburgh, PA, USA, Sep. 2012, pp. 421–430.

[14] T. O. Oshin and S. Poslad, "ERSP: An energy-efficient real-time smartphone pedometer," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Manchester, U.K., Oct. 2013, pp. 2067–2072.

[15] M. Alzantot and M. Youssef, "UPTIME: Ubiquitous pedestrian tracking using mobile phones," in *Proc. IEEE Wireless Commun. Netw. Conf. Services, Appl., Bus.*, Shanghai, China, Apr. 2012, pp. 3204–3209.

[16] I. Bylemans, M. Weyn, and M. Klepal, "Mobile phone-based displacement estimation for opportunistic localisation systems," in *Proc. 3rd Int. Conf. Mobile Ubiquitous Comput., Syst., Services Technol. (UBICOMM)*, Sliema, U.K., Oct. 2009, pp. 113–118.

[17] N. Kakiuchi and S. Kamijo, "Pedestrian dead reckoning for mobile phones through walking and running mode recognition," in *Proc. 16th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, The Hague, The Netherlands, Oct. 2013, pp. 261–267.

[18] S. Jayalath, N. Abhayasinghe, and I. Murray, "A gyroscope based accurate pedometer algorithm," in *Proc. 4th Int. Conf. Indoor Positioning Indoor Navigat.*, Montbéliard, France, Oct. 2013, pp. 510–513.

[19] Y. P. Lim, I. T. Brown, and J. C. T. Khoo, "An accurate and robust gyroscope-gased pedometer," in *Proc. 30th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBS)*, Vancouver, BC, Canada, Aug. 2008, pp. 4587–4590.

[20] J. W. Kim, H. J. Jang, D.-H. Hwang, and C. Park, "A step, stride and heading determination for the pedestrian navigation system," *J. Global Positioning Syst.*, vol. 3, nos. 1–2, pp. 273–279, 2004.

[21] A. R. Jiménez, F. Seco, C. Prieto, and J. Guevara, "A comparison of pedestrian dead-reckoning algorithms using a low-cost MEMS IMU," in *Proc. 6th IEEE Int. Symp. Intell. Signal Process. (WISP)*, Budapest, Hungary, Aug. 2009, pp. 37–42.

[22] J. A. Shockley, "Ground vehicle navigation using magnetic field variation," Ph.D. dissertation, Dept. Air Force Air Univ., Air Force Inst. Technol., Wright-Patterson AFB, OH, USA, 2012.

[23] M. Angermann, M. Frassl, M. Doniec, B. J. Julian, and P. Robertson, "Characterization of the indoor magnetic field for applications in localization and mapping," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Sydney, NSW, Australia, 2012, pp. 1–9.

[24] D. Bhatt, D. C. Sekaran, and S. R. Babu, "An intelligent framework to determine a mobile device context utilizing in-built sensors," in *Proc. Annu. IEEE India Conf. (INDICON)*, Dec. 2015, pp. 1–5.

[25] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, “Activity recognition using cell phone accelerometers,” *ACM SIGKDD Explorations Newslett.*, vol. 12, no. 2, pp. 74–82, Dec. 2010.

[26] S. Dernbach, B. Das, N. C. Krishnan, B. L. Thomas, and D. J. Cook, “Simple and complex activity recognition through smart phones,” in *Proc. 8th Int. Conf. Intell. Environ. (IE)*, Jun. 2012, pp. 214–221.

[27] A. M. Khan, Y.-K. Lee, S. Y. Lee, and T.-S. Kim, “Human activity recognition via an accelerometer-enabled-smartphone using kernel discriminant analysis,” in *Proc. 5th Int. Conf. Future Inf. Technol. (FutureTech)*, May 2010, pp. 1–6.

Arun Kumar Siddanahalli Ninge Gowda received the B.E. degree in electronics and communication engineering from Visvesvaraya Technological University, India, in 2005. He is currently with the Samsung R&D Institute India-Bangalore Pvt. Ltd., as a Technical Lead. His research interests include sensor-based pedestrian navigation, vehicle navigation, and human activity classification systems.

Swarna Ravindra Babu was with various research and development industries in both India and abroad for about 20 years. He is the Founder and the CEO of Coovum Smart Systems and Services Private Limited, Chennai. His areas of specialization are IoT, GPS/GNSS, sensors, wireless communication, RF, and statistical systems. He has developed products in the areas of GPS, sensors, autonomous navigation of agricultural tractors, and location-based services for premium smartphones. He has authored over 100 papers in technical journals and conferences, and he holds 11 patents in India and abroad. He is a reviewer of several journals including the IEEE. He has received international awards for best papers in the field of navigation. He is also an invited speaker in several conferences and workshops.

Dhineshkumar Chandra Sekaran received the B.Tech. degree in information technology from Anna University, India, in 2010. He is currently with Symbol Technologies India Pvt. Ltd., Bangalore, as a Lead Engineer. His research interests include sensor-based application development for mobile and wearable devices.