

Design and Implementation of Practical Step Detection Algorithm for Wrist-Worn Devices

Yunhoon Cho, Hyuntae Cho, and Chong-Min Kyung, *Fellow, IEEE*

Abstract—In recent years, interest in wrist-worn devices has been growing, as market of wearable activity tracking devices have been enlarged. But, many wrist-worn devices have three main problems that activity tracking algorithms for wrist-worn devices should overcome: lack of sensor variety due to power consumption, low computing power, and noise from various sensor-carrying modes and walking velocities. This paper discusses an activity tracking, especially regarding step detection algorithm using three-axis accelerometer for wrist-worn devices. The proposed algorithm consists of three phases, which address the problems of wrist-worn devices. The first data preprocessing phase calculates the Euclidean norm of the acceleration vector. It enables the algorithm to track the movement of a device only with the acceleration data. The second data filtering phase reduces the noise with a simple digital low-pass filter. Then, the third peak detection phase adopts a sign-of-slope method and average threshold method to accurately detect the step peaks under different sensor-carrying modes and velocity conditions. A wrist-worn hardware prototype is designed and realized for algorithm evaluation. The experiment results show that the proposed algorithm is superior to the compared existing algorithm and commercial devices. The averaged detection error is approximately 1% in different test conditions.

Index Terms—Activity tracking, step detection algorithm, pedometer accuracy, accelerometer, wrist-worn device.

I. INTRODUCTION

RECENTLY, wearable sensor technologies have developed rapidly and are commonly utilized in a variety of fields. Wearable devices with different sensors provide the information regarding the user's physical activities, and this information is used in many applications in fields such as fitness, security, emergency detection, and entertainment [1].

Concurrently, people's interest regarding health care and personal physical activity steadily grows, and health-care

Manuscript received July 18, 2016; revised August 19, 2016; accepted August 20, 2016. Date of publication August 25, 2016; date of current version September 28, 2016. This work was supported by the Center for Integrated Smart Sensors Funded by the Ministry of Science, ICT and Future Planning through the Global Frontier Project (CISS-2013M3A6A6073718). The associate editor coordinating the review of this paper and approving it for publication was Dr. Arindam Basu. (*Corresponding author: Hyuntae Cho.*)

Y. Cho is with the Smart Sensor Architecture Laboratory, Electrical Engineering Department, Korea Advanced Institute of Science and Technology, Daejeon 34141, South Korea (e-mail: cyh4565@kaist.ac.kr).

H. Cho is with the Center for Integrated Smart Sensors, Korea Advanced Institute of Science and Technology, Daejeon 34141, South Korea (e-mail: phd.marine@kaist.ac.kr).

C.-M. Kyung is with the Electrical Engineering Department, Korea Advanced Institute of Science and Technology, Daejeon 34141, South Korea (e-mail: kyung@kaist.ac.kr).

Digital Object Identifier 10.1109/JSEN.2016.2603163

services have consequently started to focus on “activity tracking,” whereby personal fitness-related activities are tracked and monitored in real time. Report by the Futuresource company predicts that the wearable-gadgets market, especially in fitness-related areas, will continuously expand, and a market value will reach more than US \$20 billion by 2017 [2]. The health care service trend in this regard is boosted by the development of a variety of wearable devices that provide user information through the functioning of built-in sensors. With the advancement of sensor technology (e.g., accelerometer, GPS, and gyroscope), wearable devices can now collect a user's physical activities in real time; furthermore, this information is used for various types of activity tracking such as step detection, walking distance estimation, and emergency detection [3], [4].

As the market of wearable fitness devices is rapidly growing, a significant amount of research studies on activity tracking, especially step detection, have been recently conducted [5]–[16]. Some of these studies focus on step detection or fall detection algorithms for handheld devices like the smartphone [5]–[7], and for which the machine learning technique is employed [8]. The data from three-axis accelerometer that provides the acceleration data of the accelerometer's x, y, and z axis and the gyroscope sensor are used for the detection of the step peak from a noise-added signal through the use of a number of digital filtering methods. Other researchers proposed algorithms that target the step detections of waist-worn or ankle-worn devices [9]–[11], and some of them tested the commercial wearable gadgets [12]. A research regarding general-circumstance step detection algorithms with unspecified device type is not specified has also been performed [13].

However, wrist-worn devices, which represent a rapidly rising market in the field of wearable-computing development, cannot provide similar activity tracking utility levels compared to the typical handheld, waist-worn, or ankle-worn wearable devices. The devices typically contain small numbers of sensor types and they are vulnerable to external error sources such as the sensor-carrying type or walking velocity variations. Also, it is typical for wrist-worn devices to require low computing power owing to size, heat, and battery issues. For many of the existing activity tracking algorithms, it is assumed that a tracker is worn in a stable position, and aid is received from a number of sensors including the accelerometer and the gyroscope. These algorithms also require a large amount of computing resources since complex filters and methods are utilized for their designs. Therefore, it is evident that

wrist-worn devices need an activity tracking algorithm which satisfies following conditions.

- Deals with a variety of sensing conditions
- Utilizes accelerometer data only for the detection
- Requires a low level of computing resources

In this paper, a step detection algorithm for wrist-worn wearable devices is proposed. The proposed algorithm utilizes only the data of a three-axis accelerometer, since wrist-worn devices are commonly designed with an accelerometer, while other sensors such as the GPS and the gyroscope are not frequently supported. Our algorithm is composed of a *data preprocessing phase*, a *data filtering phase*, and a *peak detection phase*. Each phase can address the conditions that algorithms for wrist-worn devices should satisfy. The data-preprocessing phase involves the calculation of the Euclidean norm of the acceleration vector, whereby the acceleration fluctuation is extracted. With the norm of acceleration vector, we can detect the entire motion of the user without the data from additional sensors like gyroscope. The data-filtering phase deals with the signal noise by adopting a simple digital low-pass filter. During the peak detection phase, the step is detected by using a combination of the sign-of-slope method and the average threshold. Existing peak detection algorithms suffer from detection failure and false detection when the device is under abnormal conditions like folded-arms sensor-carrying mode or slow walking velocity. But our peak detection phase produces accurate detection results even under abnormal sensing conditions.

To provide criteria for the evaluation of our algorithm, we test the step detection accuracy of several commercial wearable-fitness devices with different walking velocities and sensor-carrying modes: folded arms or pocketed hands for example. Furthermore, we design a wrist-worn hardware prototype and implement two algorithms, ours and the existing *Pan-Tompkins* algorithm that is proposed in [13]. We proceed with several experiments for which the hardware prototype is used to evaluate the superiority of the proposed algorithm in comparison with the others.

The contributions of this proposed algorithm are defined in terms of three aspects. First, our algorithm can be easily implemented for the design of common wrist-worn devices with low computing powers since the algorithm is composed of simple digital filters and data-processing methods, and thus has low complexity. Second, the proposed scheme provides accurate step detection results for different sensor-carrying modes and walking velocities, as shown in the experiments of Section V. Finally, our algorithm provides a high accuracy for step detection with only the use of a three-axis accelerometer, whereby other sensors like the gyroscope and the GPS are not employed. The proposed algorithm can therefore be applied to a variety of wrist-worn devices for which only the accelerometer has been loaded.

II. RELATED WORKS

A. Existing Step Detection Algorithms

As the interest regarding activity tracking continually grows, many research studies on step detection, one of the main



Fig. 1. Five filters of *Pan-Tompkins* step detection algorithm.

facets of activity tracking, have been completed. Previous step detection algorithms can be categorized according to their data-processing methods: the peak detection method, the *zero-velocity update (ZUPT)*, and the correlation-calculation method.

The peak detection method, which counts the step number by detecting the step peak of an acceleration signal, is widely used among researchers, as [7], [14], [15], and [21] show. The peak detection method is commonly used with the threshold, whereby the peak-acceleration value is checked for whether it exceeds the given threshold. This method is of a low complexity and is easy to implement, but the detection result may be affected by various external noise sources such as the user's sensor-carrying mode.

In references [6], [10], [11], [16], and [20], the authors apply the *zero-velocity update (ZUPT)* method for the detection of the user's steps. The *ZUPT* concept is based on the idea that one zero-velocity moment should occur when a user walks a single step. Algorithms for which the *ZUPT* is incorporated count the number of zero-velocity moments for the calculation of the step number.

One research study tried a correlation-calculation method for the detection of the user's steps [17]; here, the scheme is used to calculate the autocorrelation of the acceleration data with a determined start point and end point. If the autocorrelation value exceeds a certain threshold, one step is detected.

With an understanding of these step detection methods, a peak detection method of a very low complexity is selected for our algorithm, whereby filtering and threshold methods can be used to compensate for any detection errors that occur while applying the peak detection method.

B. *Pan-Tompkins* Step Detection Algorithm

Proposed in 1985 by Pan and Tompkins, the *Pan-Tompkins method* is widely known as a real-time R-peak detection algorithm for ECG signals [19]. The *Pan-Tompkins method* contains several digital filters that allow for a fast execution, and [13] suggested that these filters can be applied to a general step detection algorithm for which the acceleration signal is utilized. Specifically, the *Pan-Tompkins* step detection algorithm consists of five digital filters as shown in Fig. 1.

The first filter is a low-pass filter that reduces the influence of external noise and smoothens the signal. A digital low-pass filter with small integer coefficients for a fast execution is used for the *Pan-Tompkins method*. After filtering, the signal is differentiated through a derivative operator for the provision of slope information. In practice, this process suppresses the low-frequency components and enlarges the high-frequency components. The squaring operator, which squares the signal value point by point, functions after the differentiation

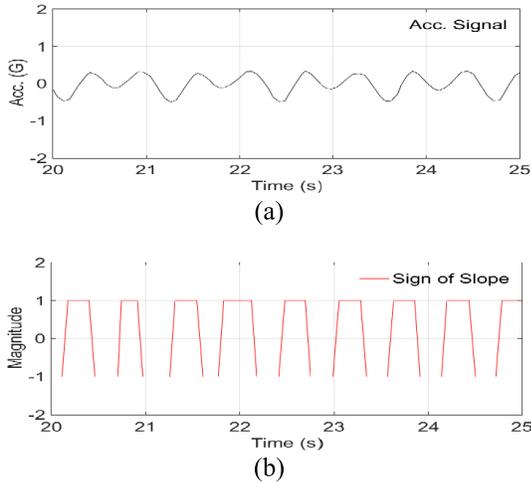


Fig. 2. Acceleration signal (a) and sign-of-slope result (b) where the unit “G” is for the gravitational acc. ($1\text{ G} = 9.8\text{m/s}^2$)

process and leads to positive signal values, whereby the large values are enhanced more than the small values to make the peak analysis easier. Since the signal after the squaring process comprises multiple peaks, a smoothing process that is achieved through a moving-window operation is needed for the improvement of the peak detection.

With the previous filters, the acceleration signal is processed for the step detection and the final process of the method, the peak detection procedure, is then completed. Various peak detection methods are used for step detection and the “simple sign-of-slope” peak detection method was selected for the *Pan-Tompkins* algorithm that is proposed in [11] to enable a fast execution. The simple sign-of-slope method processes the signal in three steps. First, it converts the signal to the bit stream of -1 , 0 , and 1 according to the signal slope; specifically, the signal is converted to 1 if the signal slope is positive, and -1 if the slope is negative. After the conversion, the number of peaks is calculated by counting $[1\ -1]$ since the upward and downward slopes define the peak. Figure 2 shows the results of the sign-of-slope method.

Based on these discussions, we propose a step detection algorithm for wrist-worn devices for which only three-axis accelerometer data is used; furthermore, the algorithm can be used to accurately count the step number at different walking velocities and under varying sensor-carrying mode circumstances. Since the *Pan-Tompkins* algorithm is a general algorithm for which the device type is not considered, our algorithm and the *Pan-Tompkins* algorithm are tested in terms of the step detection accuracy; as a result, our algorithm shows a step detection accuracy that is greater than that of the *Pan-Tompkins* algorithm.

III. ACTIVITY TRACKING ALGORITHM

In this section, we propose a step detection algorithm for wrist-worn device, which is composed of three phases: data preprocessing, data filtering, and peak detection. Fig. 3 briefly shows the executions of three phases.

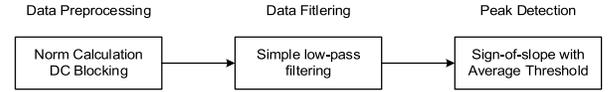


Fig. 3. Three processes of our step detection algorithm.

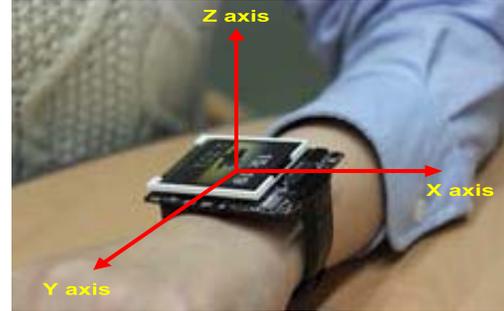


Fig. 4. The three-axis coordinate of the wrist-worn device.

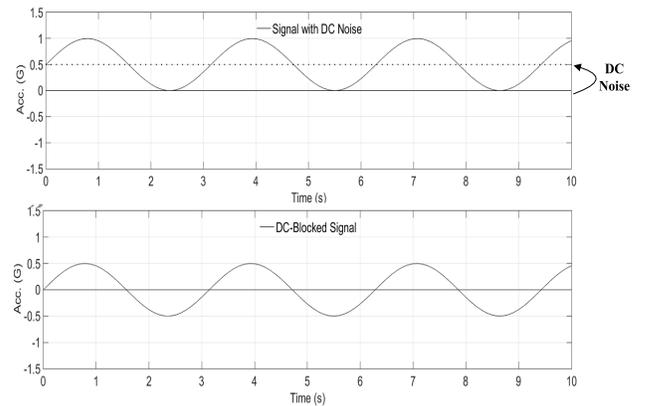


Fig. 5. DC or fixed noise blocking process.

A. Data Preprocessing Phase

In this phase, raw three-axis acceleration data is gathered from a wrist-worn device with a pre-determined sampling rate. For the wrist-worn device, the coordinate system is defined relative to the accelerator chip as shown in Fig. 4. At time t_i , the sensed acceleration vector and the values from the three-axis accelerometer are defined as $\vec{A}(t_i) = (a_x(t_i), a_y(t_i), a_z(t_i))$. After the acquisition of $\vec{A}(t_i)$, the data is preprocessed using the Euclidean norm calculation $\|\vec{A}(t_i)\| = u_{acc}(t_i)$, and DC or fixed noise blocking. Fig. 5 shows the DC blocking process.

A user's walking causes arm movements and this movement is transferred to the wrist-worn device. The variation of the sensed acceleration values from the wrist-worn device therefore represents the movement of the device allowing for the detection of the user's walking. The raw acceleration values from the device should be processed with the use of several methods since the values do not directly represent the actual walking of the user and this processing is performed during the data-preprocessing phase. Specifically, when a user walks arbitrarily with a wrist-worn device, the information of the user's movement is distributed through the acceleration

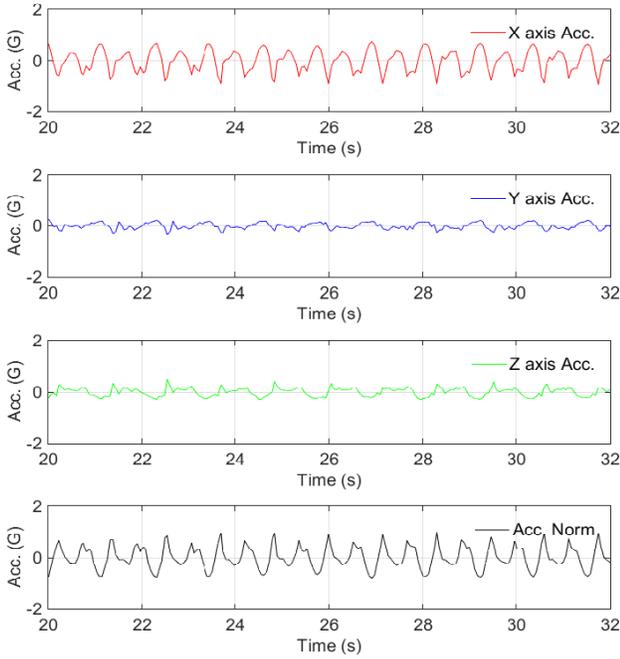


Fig. 6. X-, Y-, and Z-axes acceleration data and their DC-blocked norm.

of the X, Y, and Z axes; therefore, one should observe the acceleration variations of all three axes, or the acceleration vector $\vec{A}(t)$ to track the user's entire movements rather than using the acceleration data of one or two axes.

Several methods can be used for the gathering of the entire information regarding the acceleration of the three axes, such as the coordinate rotation of [3] or the use of a gyroscope for device-orientation detection, but the angle information of the device is needed for a calculation of the orientation of the sensor, whereby a relatively high computing power and extra sensors are required. Therefore, we propose the concept of the Euclidean norm, shortened to "norm," regarding the acceleration vector for a fast and simple detection of the movement of the device that is not dependent on angle or orientation information. The definition of the Euclidean norm is as follows: Assume the existence of a three-dimensional Euclidean space vector $\vec{A} = (a, b, c)$. To evaluate the size of the vector, a one-dimensional factor should represent the size and the Euclidean norm is this factor. The formula of the Euclidean norm is as follows:

$$\|\vec{A}\| = \sqrt{a^2 + b^2 + c^2}$$

For our algorithm, the Euclidean norm of the acceleration vector $\vec{A}(t_i)$, which varies as the user walks, is calculated as follows:

$$\|\vec{A}(t_i)\| = u_{acc}(t_i) = \sqrt{a_x(t_i)^2 + a_y(t_i)^2 + a_z(t_i)^2}$$

Notably, a nonzero DC component that hides the step information and reduces the accuracy of the peak detection procedure can be relevant here. We therefore adopt the DC-blocking process for the norm $u_{acc}(t_i)$ by subtracting the moving

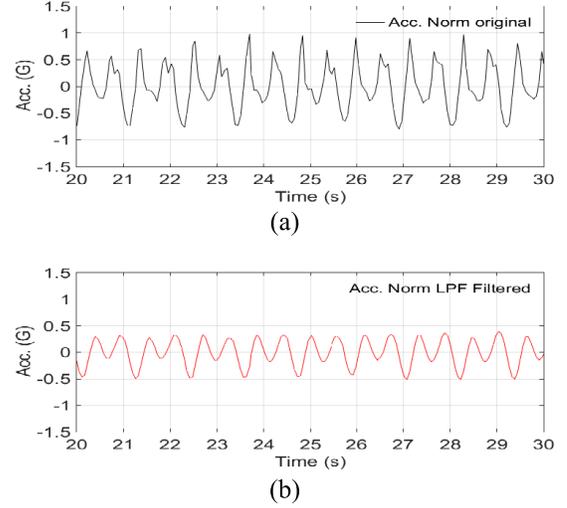


Fig. 7. Original acceleration norm (a) and low-pass filtered norm (b).

average of the norm from the original norm as follows:

$$u_{acc,0}(t_i) = u_{acc}(t_i) - \frac{1}{N} \sum_{k=i-N+1}^i u_{acc}(t_k)$$

The moving of the average window size N is empirically determined as 20.

We track the variation of the DC-blocked Euclidean norm of the acceleration vector $u_{acc,0}(t_i)$ for the detection of the user's walking. Figure 6 shows the acceleration signals of the X, Y, and Z axes and their DC-blocked norm.

B. Data Filtering Phase

The acceleration data from a wrist-worn device is vulnerable to the noise from external circumstances such as the sensor-carrying modes of the user. This signal noise affects the calculated norm and the peak detection that leads to the filtering of the norm in the data-filtering phase.

For a fast execution and a low-complexity method, we adopt a digital low-pass filter for the data filtering phase. Human-movement information occurs under a frequency of 20 Hz [22]. So a low-pass filter with a cutoff frequency of 20 Hz is designed for this phase. The transfer function $H(z)$ and the difference equation $y[n]$ of the filter are as follows.

$$H(z) = \frac{1}{16} \frac{(1 - z^{-4})^2}{(1 - z^{-1})^2}$$

$$y[n] = \frac{1}{16} (x[n] + 2x[n-1] + 3x[n-2] + 4x[n-3] + 3x[n-4] + 2x[n-5] + x[n-6])$$

In these equations, "z" is the complex frequency of the system in frequency domain. "x" is the system input in discrete time domain. And "n" is the time in discrete time domain.

Fig. 7 shows the input and output of data filtering phase. Fig. 7 (a) is DC blocked acceleration norm, $u_{acc,0}$, and Fig. 7 (b) is low-pass filtered norm, which is the output of data filtering phase. After the data filtering phase, low-pass filtered

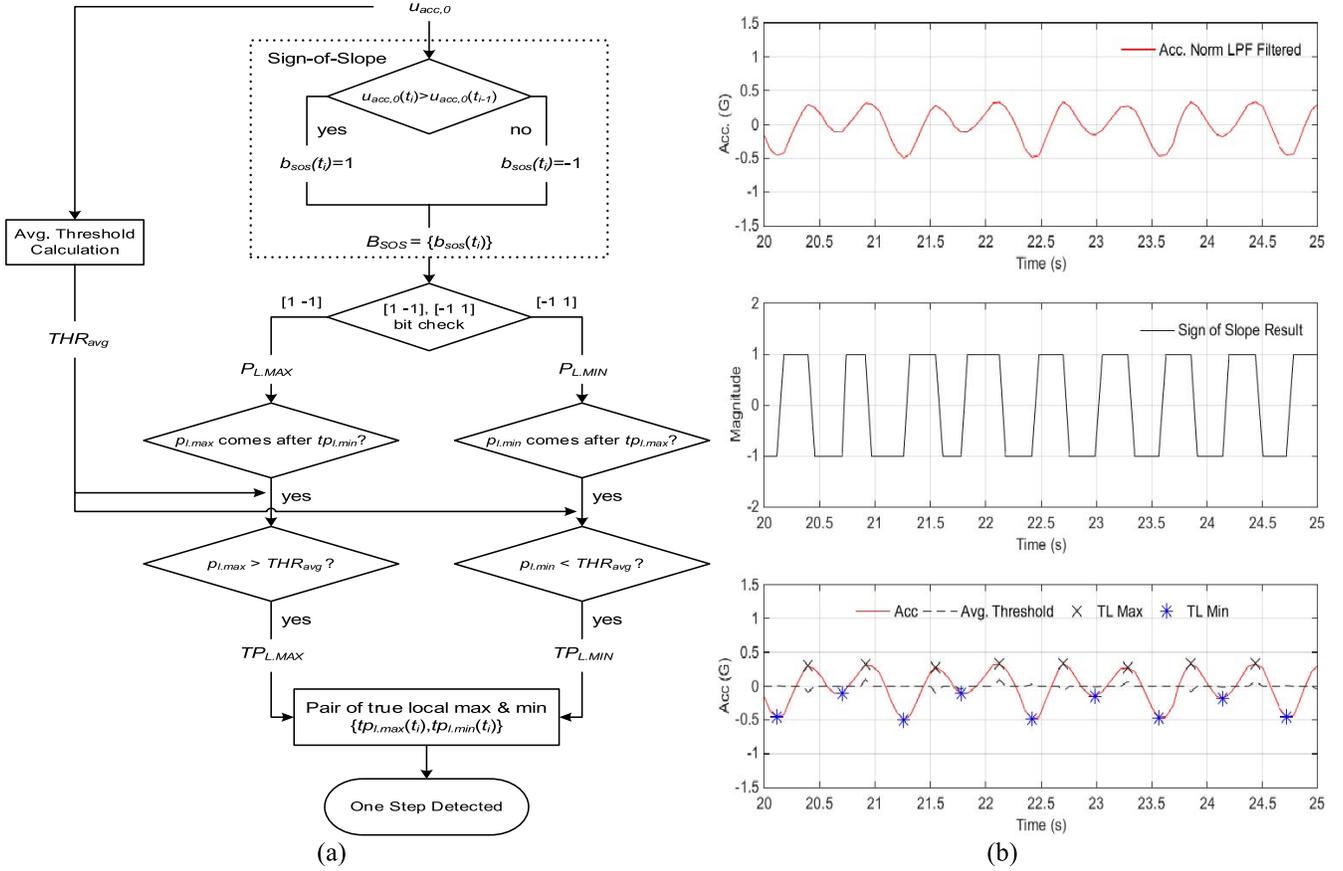


Fig. 8. Flow chart of the peak detection phase (a) and signal processing result of peak detection phase (b).

norm shows smoother signal and the step peaks become more significant than $u_{acc,0}$.

C. Peak Detection Phase

In Section II, the different step detection methods, such as ZUPT and autocorrelation-matching, are discussed. But the algorithm complexity of these methods is relatively high and they are therefore not suitable for the type of wrist-worn device that is targeted in this study; therefore, the peak detection phase is included for the step detection process. A peak detection method for which the sign-of-slope method and the average threshold are used for the achievement of both a fast execution and the compensation of the signal noise after the low-pass filtering is proposed in this paper.

The basic idea of our peak detection method is as follows: The filtered acceleration norm is converted to the bit stream of 1 and -1 , whereby $B_{SOS} = \{b_{sos}(t_i)\}$, through the sign-of-slope process. In specific, bit $b_{sos}(t_i)$, slope of the norm signal at time t_i , has a value of 1 if the slope is positive and -1 if the slope is negative. Thus a couple of bits $[b_{sos}(t_i) = 1 \ b_{sos}(t_{i+1}) = -1]$ represent the local maximum, and $[b_{sos}(t_i) = -1 \ b_{sos}(t_{i+1}) = 1]$ represent the local minimum as Fig. 8(b) shows. Therefore, the candidates of the local maximum and local minimum of the norm are then found by searching the $[1 \ -1]$ bit that represents the local

maximum, and the $[-1 \ 1]$ bit that represents the local minimum in B_{SOS} . After the lists of candidates for the local maximum, $P_{L,MAX} = \{p_{l,max}(t_i)\}$, and the local minimum, $P_{L,MIN} = \{p_{l,min}(t_i)\}$, are made, the average threshold $THR_{avg}(t_i)$ of the norm that is determined by the moving average of the norm values is calculated. The candidates, $p_{l,max}(t_i)$ and $p_{l,min}(t_i)$, are then evaluated for whether they are the true local maximum and the true local minimum through the checking of the condition—the true local maximum should come after the true local minimum, and vice versa—and the values are then compared with $THR_{avg}(t_i)$. The number of steps are detected by counting the number of either the true local maximum, $TP_{L,MAX} = \{t_{p,l,max}(t_i)\}$, or the true local minimum, $TP_{L,MIN} = \{t_{p,l,min}(t_i)\}$.

We now explain each of the steps of the peak detection phase more specifically.

First, the filtered acceleration norm passes the sign-of-slope process and the stream of $[1 \ -1]$ bits are produced according to the slope of the given norm; specifically, the sign-of-slope bit $b_{sos}(t_i) = 1$ if $u_{acc,0}(t_i) > u_{acc,0}(t_{i-1})$ and $b_{sos}(t_i) = -1$ if $u_{acc,0}(t_i) < u_{acc,0}(t_{i-1})$.

The $[1 \ -1]$ bit therefore represents the local maximum of the signal and the $[-1 \ 1]$ bit represents the local minimum; however, these detected local maximums and local minimums may not be true representations, and false detection can occur during the bit-check process as shown in Fig. 9. These detected

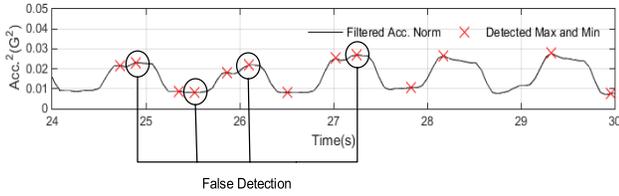


Fig. 9. False detection during [1 -1] and [-1 1] bit check.

local maximums and local minimums are therefore set from the bit-check process regarding the candidates for the true local maximums and local minimums, $P_{L,MAX} = \{p_{l,max}(t_i)\}$ and $P_{L,MIN} = \{p_{l,min}(t_i)\}$, and they are evaluated to determine the true peaks according to the following two criteria:

- 1) The true local maximum should come after the true local minimum, and vice versa: If $p_{l,max}(t_{i+1})$ comes after $tp_{l,max}(t_i)$ and not $tp_{l,min}(t_i)$, $p_{l,max}(t_{i+1})$ cannot be a true local maximum, $tp_{l,max}(t_{i+1})$.
- 2) The true local maximum should exceed the average threshold and the true local minimum should be less than the average threshold: $tp_{l,max}(t_i) > THR_{avg}(t_i)$ and $tp_{l,min}(t_i) > THR_{avg}(t_i)$.

The average threshold of the filtered norm, $THR_{avg}(t_i)$, is calculated according to the moving average of the norm with an empirically determined window size of $N = 10$ as the following formula shows:

$$THR_{avg}(t_i) = \frac{1}{N} \sum_{k=i-N+1}^i u_{acc,0}(t_k)$$

If the candidates, $P_{L,MAX}$ and $P_{L,MIN}$, satisfy the two criteria, they are selected as the true local maximums and local minimums, $TP_{L,MAX}$ and $TP_{L,MIN}$. And the number of steps is finally determined by the counting of the number of pairs of the true local maximum and the true local minimum. Figure 8 shows a flow chart of the peak detection phase (a) and the peak detection result of the filtered norm signal (b).

IV. PROTOTYPE OF THE SYSTEM

Figure 10 shows the step detection device that was developed for this study, where (a) illustrates the block diagram of the system and (b) is the hardware prototype. This device measures personal activity such as steps, health, and environmental-contamination exposure. The device uses an ST Microelectronics STM32f4xx (ARM Cortex-M4) chip that runs the FreeRTOS operating system. This micro-controller unit (MCU) provides a sound performance and convenient functions for our purposes. We added a FAT32 file system [18] on this MCU to record the measured data and the event history. To measure acceleration, the device uses the Analog Device ADXL362 three-axis accelerometer, which is an ultra-low-power IC that consumes $1.8 \mu A$ at a 100 Hz output-data rate (ODR) with a 2.0 V supply. We set the ODR to 20 Hz to greatly reduce the power consumption. Measurement ranges of $\pm 2 g$, $\pm 4 g$, and $\pm 8 g$ are available with a resolution of 1 mg/LSB on the $\pm 2 g$ range. We set the basic gravity to $\pm 2 g$.

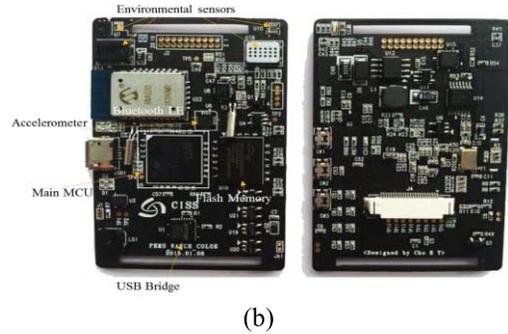
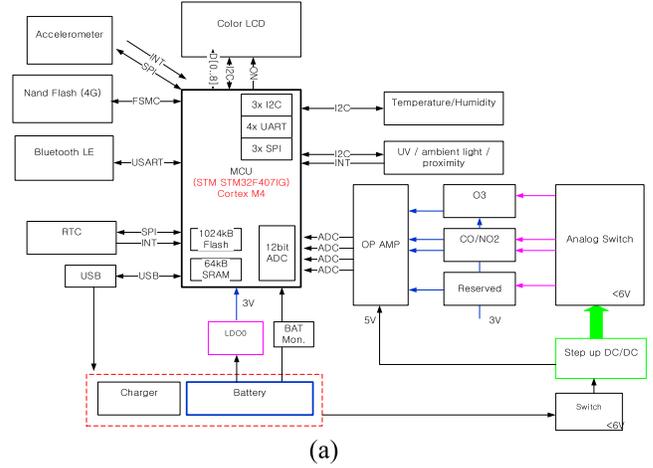


Fig. 10. Wrist-worn device for step detection: (a) block diagram and (b) hardware prototype.

The device includes other components such as sensors, a power-management IC, and a communication module for the provision of further functions as shown in Figure 10 (a). It contains two digital sensors (temperature/humidity and UV-/ambient-light sensing) and four analog sensors (O_3 , NO_x , CO_x , and VOC_s) for the measurement of ambient air pollution. Each sensor has a small form factor and consumes less energy than the traditional wearable-device sensors. A 2.8-inch LCD display with a capacitive touch panel is also part of the design for the display of the device's status and other relevant information. The sensed data through the device is transferred to the Internet via a smartphone or a gateway. The device uses a BLE module (Microchip RN4020) that consumes a low amount of energy (17 mA @ Tx) when the sensing data is transmitted. This device is used to extract the acceleration data and for the evaluation of the proposed method. We deal again with this device and the corresponding experiments in Section V.

V. EXPERIMENTS AND PERFORMANCE EVALUATION

A. Experiments for Step Detection Algorithm

For this paper, a simple step detection algorithm for wrist-worn devices is formulated and implemented on a wrist-worn prototype device. To evaluate the accuracy of our algorithm, we proceed with a number of treadmill experiments for which the prototype hardware and commercial pedometers are used.

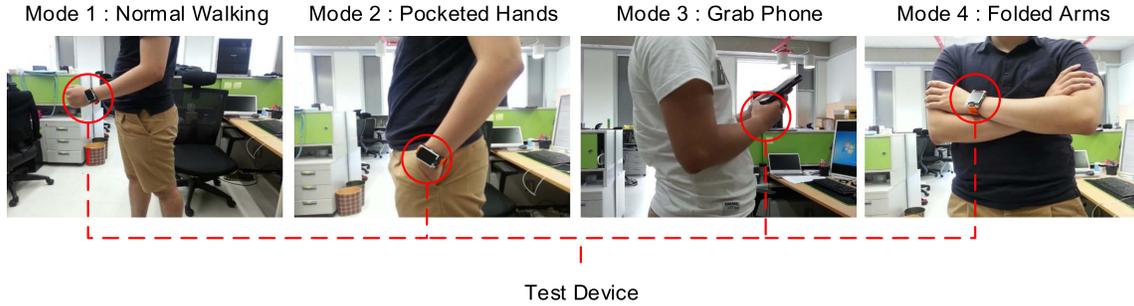


Fig. 11. Four tested sensor carrying modes.



Fig. 12. Tested commercial pedometers.

As mentioned previously, a wrist-worn device is commonly exposed to a variety of conditions that vary according to the user's sensor-carrying mode or velocity. We therefore select four major sensor-carrying modes and four walking velocities that represent a variety of circumstances that such a device is likely to be exposed to. The four sensor-carrying modes, as shown in Fig. 11, are as follows:

1) Normal Walking

This mode is the basic sensor-carrying mode for which the user walks with normally swinging arms. The device experiences a pendular movement and the acceleration-norm signal reveals the step peak clearly.

2) Pocketed Hands

This mode refers to the case where the user walks with both hands in pockets. In this case, the step peaks of the signal might be smaller than those of mode 1.

3) Grab Phone (Hand Texting)

This mode includes the case where the user walks while grabbing a handset or texting a message on the handset. In this case, the device is almost stationary.

4) Folded Arms

This mode refers to the case where the user folds his/her arms. The device is also quasi-stationary in this case.

The four selected walking velocities are as follows:

- 1) 2.5km/h: Slow walking
- 2) 4.5km/h: Normal walking
- 3) 6.5km/h: Fast walking
- 4) 8.5km/h: Normal running

If the user moves faster, the step peaks become larger and close to each other. And in the case of running, the growth of the step peaks is very large.

We proceed with the treadmill experiments for which the commercial pedometers and our prototype hardware are used by changing the previously mentioned factors. Through the performance of these experiments, it is possible to determine the accuracy of the algorithm when the device is exposed to various error sources.

At first, we set the step detection accuracy as a criterion by measuring the step detection accuracy with the selected commercial pedometers. The current commercial pedometers can be classified into the following three types: smartwatch, smartband, and handheld device. And specific devices are selected for each type. Figure 12 shows the variety of tested commercial devices.

The step detection accuracy measurements of the commercial pedometers are undertaken during a treadmill experiment. With the use of different sensor-carrying modes and velocities, 500 steps are completed on the treadmill through walking or running, and the detection error is calculated according to the following formula:

$$\text{Error (\%)} = 100 * \frac{|Detected Step Number - 500(steps)|}{500(steps)}$$

To accurately compare all commercial devices, we proceed with the experiments wherein all of the devices are worn at once. We walk on the treadmill at 4.5 km/h when we test the step detection accuracies with different sensor-carrying modes, and we walk with the sensor-carrying mode 1 when we tested the walking velocity variation case.

In terms of the prototype experiments, a different protocol is set for the purpose of convenience. We walk for a 120 seconds on the treadmill with different sensor-carrying modes and velocities. While undertaking the experiments, the hardware prototype is worn on our wrists and the ground-truth step numbers are counted.

Figure 13 shows the test conditions of the commercial pedometers and our hardware prototype.

B. Step Detection Accuracy With Different Sensor-carrying Modes and Velocity Circumstances

The step detection accuracy of our algorithm is evaluated through its implementation in the design of our wrist-worn hardware prototype. We also implement the existing general step detection *Pan-Tompkins* algorithm for a comparison.



Fig. 13. Test conditions of commercial pedometers (a) and our hardware prototype (b).

TABLE I
EXPERIMENT RESULTS OF STEP DETECTION ERROR WITH DIFFERENT SENSOR-CARRYING MODES

Algorithm Type	Mode 1 Normal Walking	Mode 2 Pocketed Hands	Mode 3 Grab Phone	Mode 4 Folded Arms	Averaged Error
<i>Pan-Tompkins</i> Algorithm	0.41% \pm 0.42%	23.1% \pm 10.6%	33.6% \pm 9.98%	38.9% \pm 30.0%	24.00% \pm 12.75%
Our Algorithm	0.76% \pm 0.60%	1.08% \pm 0.85%	0.83% \pm 0.79%	1.24% \pm 0.66%	0.98% \pm 0.73%

TABLE II
EXPERIMENT RESULTS OF STEP DETECTION ERROR WITH DIFFERENT WALKING VELOCITIES

Algorithm Type	2.5km/h Slow Walking	4.5km/h Normal Walking	6.5km/h Fast Walking	8.5km/h Normal Running	Averaged Error
<i>Pan-Tompkins</i> Algorithm	15.1% \pm 8.54%	0.41% \pm 0.42%	28.2% \pm 10.8%	36.0% \pm 15.8%	19.93% \pm 8.89%
Our Algorithm	0.89% \pm 0.61%	0.76% \pm 0.60%	2.04% \pm 1.14%	1.16% \pm 0.75%	1.21% \pm 0.78%

The step detection error is calculated using 20 test data from the treadmill experiments.

From Table I, it is evident that our algorithm outperforms the *Pan-Tompkins* algorithm through its revelation of the detection error at around 1 % in all four of the sensor-carrying modes. The *Pan-Tompkins* algorithm shows the step detection error at around 0.5 % in mode 1, but the detection errors in modes 2, 3, and 4 all exceed 20 %.

This detection error tendency is also shown in the walking-velocity-variation experiments. Table II shows the experiment results with different walking velocities. From the results, we can see that the detection error of our algorithm is around 1 % in all four velocities; in contrast, the performances of the *Pan-Tompkins* algorithm for slow walking (2.5 km/h), fast walking (6.5 km/h), and normal running (8.5 km/h) are problematic with detection errors of more than 15 %. The detection accuracy of the *Pan-Tompkins* algorithm for normal walking (4.5 km/h), however, is sound. Since a simple low-pass filter is adopted for data filtering with respect to our algorithm, and the steps are detected through the peak detection method that might produce a significant error, our algorithm could show a high detection error when the device is under different sensor-carrying modes or walking velocities. However, the experiment result shows that our algorithm can deal with the detection noises from different sensor-carrying modes and walking velocities. The high detection accuracy of the proposed algorithm in various test conditions is from the improved peak detection phase. It is clear that abnormal sensor-carrying modes and

walking velocities can cause unclear step peaks and signal noises. And existing peak detection methods cannot deal with these problems properly. Therefore, they are vulnerable to sensor-carrying mode or walking velocity variations. However, our peak detection phase can compensate the signal noises through average threshold. Also, two criteria for true step peaks enables the proposed algorithm to detect the unclear step peaks correctly.

The detection accuracy degradation of the *Pan-Tompkins* algorithm under various conditions can be explained by its signal waveform after the filtering process. Figure 14 shows the signal waveforms after our filtering phase and the *Pan-Tompkins* filters with different sensor-carrying modes and walking velocities. In “mode 1: normal walking,” the signals after our filtering phase and the *Pan-Tompkins* filters both reveal the step peak, and step detection errors of less than 1 % are consequently produced; however, the signals from the *Pan-Tompkins* filters in modes 2, 3, and 4 reveal distorted step peaks that are affected by noises. The appearance of these peaks is due to a reduction of the signal power by the filters of the *Pan-Tompkins* algorithm, whereby the simple sign-of-slope peak detection phase for which a proper compensation or noise-recognition method does not apply cannot find the difference from the real step peaks and signal noises as a result. These ambiguous step peaks cause a false detection and the *Pan-Tompkins* algorithm therefore produces a detection error of more than 20 %; moreover, the *Pan-Tompkins* algorithm shows its weakness in the 2.5 km/h, 6.5 km/h, and 8.5 km/h cases. The *Pan-Tompkins* algorithm also shows

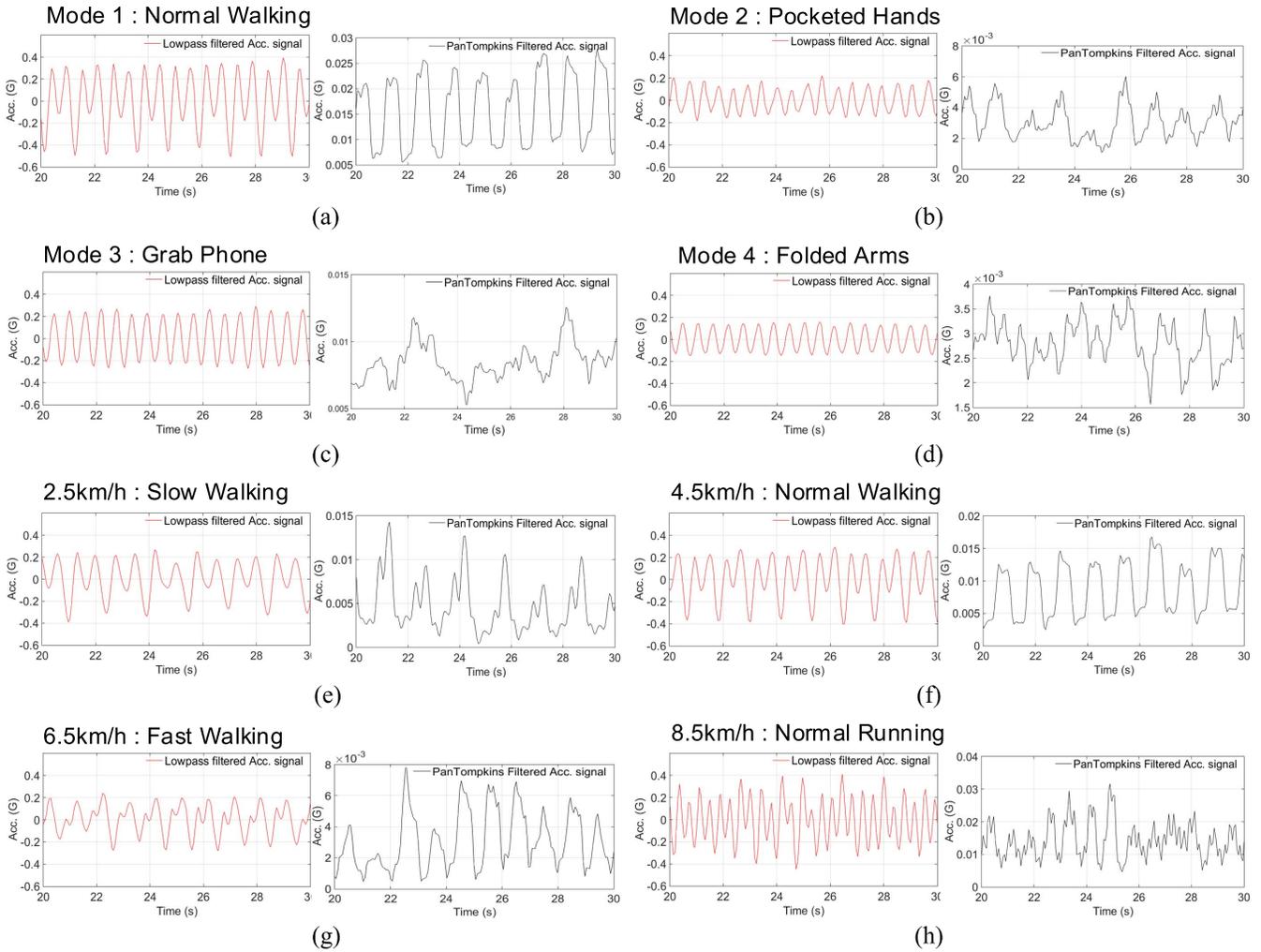


Fig. 14. Signals after our filtering process and *Pan-Tompkins* filters with different sensor-carrying modes and velocities.

a poor detection performance with detection errors of more than 25 % in the fast-walking and normal-running cases. While the reason for this accuracy degradation seems to again be the signal waveform after the filtering process, it is also caused by the reduced distances between the step peaks when the user moves quickly, whereby the filters of the *Pan-Tompkins* algorithm cannot separate the close step peaks and the step peaks are eventually merged through the filtering process. In the “2.5 km/h: slow walking” case, the cause of the high detection error seems to be a false detection from the use of the sign-of-slope peak detection method.

Regarding the comparison between our algorithm and the general *Pan-Tompkins* algorithm, we also tested the step detection performance of our algorithm by using the protocols that were explained earlier in this paper to acquire the detection-error data of the commercial pedometer devices. The step detection error was calculated by using 20 test data from the treadmill experiments.

Figure 15 shows the step-detection-error chart of the commercial devices and our algorithm with (a) different sensor-carrying modes and (b) walking velocities. The results show that the detection errors of the proposed algorithm

are around 1 % in all of the cases, which means that the commercial pedometers are outperformed.

From the results of Fig. 15 (a), the SKT Smartband shows the worst average detection accuracy in terms of the different sensor-carrying modes among the tested wrist-worn devices. The SKT Smartband showed a relatively effective sensitivity for the detection of steps while the user walks normally, but it produces a detection error of more than 15 % if the user walks with different sensor-carrying modes that produce step peaks that are smaller than those of mode 1; therefore, it is reasonable to conclude that the SKT Smartband cannot detect steps accurately if the motion of the device is smaller than that of the normal case. Regarding the other wrist-worn devices such as the Motorola moto360 and the Samsung Gear S, we observed good detection performances for all four of the carrying modes, but carrying modes for which the detection errors are more than 2.5 % were also identified here; for example, the Samsung Gear S shows a detection error of around 1 % for modes 2, 3, and 4, while its performance is degraded to 7.3 % in mode 1. The experiments show that the Samsung Gear S can properly detect steps when the device is quasi-static, but it cannot accurately detect the steps of a user who swings his/her arms. The iPhone 6 *Health* shows

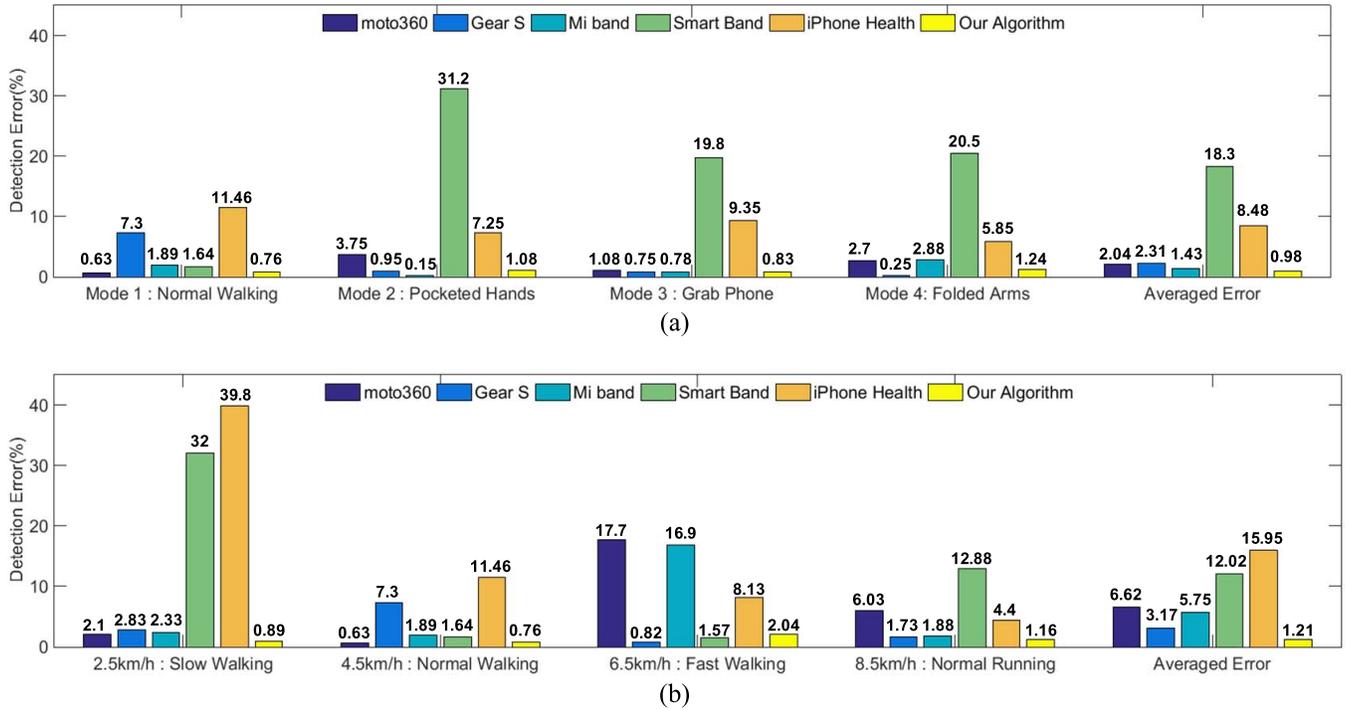


Fig. 15. Step detection errors of commercial devices and our algorithm with (a) different sensor-carrying modes and (b) walking velocities.

relatively high detection errors, which are around 10 %, in all four of the modes.

Figure 15 (b) shows the detection accuracy results of the commercial devices at different walking velocities. The SKT Smartband and iPhone 6 *Health* show high detection errors for the “2.5 km/h: slow walking” case. These results make sense since the step peaks are small in the slow-walking case and difficulties are encountered regarding the detection of small step peaks, as the sensor-carrying-mode experiments show. The Motorola moto360 and Xiaomi Mi band show high detection errors for the fast-walking case of 6.5 km/h. The iPhone 6 *Health* shows a higher detection accuracy if the user moves faster, and the average accuracy of the four velocity cases is 15.95 %, which is the worst performance among the commercial pedometers.

C. The Trade-off Between Using Additional Sensors and Power Consumption

As we mentioned in the previous part of the paper, wrist-worn devices commonly load only three-axis accelerometer due to the power consumption. And the proposed algorithm targets the devices that loads the accelerometer only. However, it is evident that the detection accuracy may go up with the aid of additional sensors, such as gyroscope or magnetometer. Therefore, the analysis of the tradeoff between using additional sensors and power consumption may be needed.

To check the benefit of using additional sensors regarding detection accuracy, we have searched the studies on the step detection using additional sensors, especially gyroscope. And, [6] and [9] matched the condition. In [6], the averaged detection accuracy was 99.2%. And the averaged detection accuracy of [9] was 99.5%. Considering that our algorithm has an averaged detection accuracy of 99% in different

TABLE III
POWER CONSUMPTION OF IMUS (ACC. AND GYRO.)

Manufacturer	Product Num.	Power Consumption
Invensense	MPU-9250	6.8mW
Invensense	ICM-20648	2.54mW
ST	LSM9DS1	10.12mW
ST	LSM6DS33	1.8mW
Bosch-Sensortec	BMI160	2.22mW

sensor-carrying modes and walking velocities, using additional gyroscope does not result in significant improvement on detection accuracy.

However, the power consumption of the gyroscope modules is relatively high. We have searched the power consumption of several commercial IMUs that have both accelerometer and gyroscope modules as table III shows. In contrast, the accelerometer module which we used, AD ADXL362, has a power consumption of $20\mu\text{W}$. Therefore, it is certain that the power consumption rises drastically when we use additional sensors for the detection.

With these discussions, it can be summarized that using additional sensors for wrist-worn device has not much benefit.

VI. CONCLUSION

In this paper, we have suggested a practical step detection algorithm for wrist-worn devices. A step detection algorithm for a wrist-worn device needs to be of a low complexity, and accurate detection results need to be produced for the various sensor-carrying modes and velocity circumstances, since the computing power of wrist-worn devices is typically low and they are exposed to various noise sources.

The proposed algorithm is composed of the following three main phases: data preprocessing phase, data filtering phase, and peak detection phase. The acceleration data from a three-axis accelerometer were transformed into a norm value, and the DC data are blocked in the preprocessing phase before they are filtered through a simple digital low-pass filter. Lastly, the sign-of-slope method is used to apply an adaptive threshold to the filtered data for the detection of the step peak. We evaluated our algorithm, including a comparison of its step detection accuracy with those of the commercial pedometers and an existing general step detection algorithm for which our wrist-worn hardware prototype was used. The experiment results indicate that the proposed algorithm outperforms other commercial devices and the *Pan-Tompkins* algorithm in terms of the step detection accuracy for all four sensor-carrying modes and all four walking velocities. In the future, we will use the results of this paper as the basis for research regarding sensor-carrying mode detection and walking-distance estimation.

REFERENCES

- [1] S. C. Mukhopadhyay, "Wearable sensors for human activity monitoring: A review," *IEEE Sensors J.*, vol. 15, no. 3, pp. 1321–1330, Mar. 2015.
- [2] *Worldwide Wearable Technology Market Report: November 2013*, accessed on Sep. 19, 2014. [Online]. Available: <http://www.future-source consulting.com>
- [3] L. Tong, Q. Song, Y. Ge, and M. Liu, "HMM-based human fall detection and prediction method using tri-axial accelerometer," *IEEE Sensors J.*, vol. 13, no. 5, pp. 1849–1856, May 2013.
- [4] T. Shany, S. J. Redmond, M. R. Narayanan, and N. H. Lovell, "Sensors-based wearable systems for monitoring of human movement and falls," *IEEE Sensors J.*, vol. 12, no. 3, pp. 658–670, Mar. 2012.
- [5] M. S. Pan and H. W. Lin, "A step counting algorithm for smartphone users: Design and implementation," *IEEE Sensors J.*, vol. 15, no. 4, pp. 2296–2305, Apr. 2015.
- [6] M. Susi, V. Renaudin, and G. Lachapelle, "Motion mode recognition and step detection algorithms for mobile phone users," *Sensors*, vol. 13, no. 2, pp. 1539–1562, Jan. 2013.
- [7] M. Mladenov and M. Mock, "A step counter service for Java-enabled devices using a built-in accelerometer," in *Proc. ACM Int. Workshop Context-Aware Middleware Services (CAMS)*, 2009, pp. 1–5.
- [8] H. Zhang, W. Yuan, Q. Shen, T. Li, and H. Chang, "A handheld inertial pedestrian navigation system with accurate step modes and device poses recognition," *IEEE Sensors J.*, vol. 15, no. 3, pp. 1421–1429, Mar. 2015.
- [9] J. W. Kim, H. J. Jang, D. H. Hwang, and C. S. Park, "A step, stride and heading determination for the pedestrian navigation system," *J. Global Positioning Syst.*, vol. 3, nos. 1–2, pp. 273–279, Nov. 2004.
- [10] O. Bebek *et al.*, "Personal navigation via shoe mounted inertial measurement units," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2010, pp. 1052–1058.
- [11] S. H. Shin, C. G. Park, J. W. Kim, H. S. Hong, and J. M. Lee, "Adaptive step length estimation algorithm using low-cost MEMS inertial sensors," in *Proc. IEEE Sensors Appl. Symp. (SAS)*, Feb. 2007, pp. 1–5.
- [12] F. Guo, Y. Li, M. S. Kankanhalli, and M. S. Brown, "An evaluation of wearable activity monitoring devices," in *Proc. 1st ACM Int. Workshop Pers. Data Meets Distrib. Multimedia Meets*, Oct. 2013, pp. 31–34.
- [13] H. Ying, C. Silex, A. Schnitzer, S. Leonhardt, and M. Schiek, "Automatic step detection in the accelerometer signal," in *Proc. 4th Int. Workshop Wearable Implant. Body Sensor Netw. (BSN)*, 2007, pp. 80–85.
- [14] K. Tumkur and S. Subbiah, "Modeling human walking for step detection and stride determination by 3-axis accelerometer readings in pedometer," in *Proc. IEEE 4th Int. Conf. Comput. Intell., Modeling Simulation (CIM-SiM)*, Sep. 2012, pp. 199–204.
- [15] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: Unsupervised indoor localization," in *Proc. 10th Int. Conf. Mobile Syst., Appl., Services (MobiSys)*, 2012, pp. 197–210.
- [16] C.-C. Lo, C.-P. Chiu, Y.-C. Tseng, S.-A. Chang, and L.-C. Kuo, "A walking velocity update technique for pedestrian dead-reckoning applications," in *Proc. IEEE 22nd Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Sep. 2011, pp. 1249–1253.
- [17] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: Zero-effort crowdsourcing for indoor localization," in *Proc. 18th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2012, pp. 293–304.
- [18] C.-C. Yang and Y.-L. Hsu, "A review of accelerometry-based wearable motion detectors for physical activity monitoring," *Sensors*, vol. 10, no. 8, pp. 7772–7788, 2010.
- [19] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm," *IEEE Trans. Biomed. Eng.*, vol. 32, no. 3, pp. 230–236, Mar. 1985.
- [20] L. Ojeda and J. Borenstein, "Non-GPS navigation with the personal dead-reckoning system," in *Proc. Defense Secur. Symp. (DSS)*, Apr. 2007, pp. 1–11.
- [21] H.-J. Jang, J. W. Kim, and D.-H. Hwang, "Robust step detection method for pedestrian navigation systems," *Electron. Lett.*, vol. 43, no. 14, pp. 749–751, Jul. 2007.
- [22] G. Bieber and A. Thom, "DiaTrace—Neuartiges Assistenz-System für die Gesundheitsprävention zur Nahrungsaufnahme und Bewegungserfassung," in *Proc. Deutscher Kongress Mit Ausstellung/Technologien-Anwendungen-Management*, Berlin, Germany, 2008, pp. 275–280.



Yunhoon Cho received the B.S. degree in electrical engineering from the Korea Advanced Institute of Science and Technology, Daejeon, South Korea, in 2015, where he is currently pursuing the Ph.D. degree. His research interests include activity tracking algorithms, wireless embedded systems, healthcare IC, and wearable computing.



Hyuntae Cho received the B.S. degree from National Korea Maritime University, Busan, in 2003, and the M.S. and Ph.D. degrees from Pusan National University, Busan, South Korea, in 2005, and 2011, respectively. He has been a Research Professor with the Center for Integrated Smart Sensors since 2012. He has developed a number of sensor-measurement devices, and is currently developing personal environmental-monitoring systems and the corresponding network platforms. He was a Research Professor with the Institute of Logistics

Information Technology, Pusan National University, from 2011 to 2012. He was an Expert Member of the Research and Development Department for Medical Devices with the Ministry of Food and Drug Safety from 2014 to 2015. He has authored 15 journal papers, 18 conference papers, and 35 patents where he is the first author. His research interests include active RFIDs, real-time locating systems (RTLS), precision time synchronization, wireless embedded systems, and wireless environment-monitoring systems. He received a prize for excellence from BK21 at Pusan National University in 2007 and awarded by the Minister of Science, ICT and Future Planning, South Korea, in 2013. He is listed in *Marquis Who's Who in the World* (36th edition) in 2016.



Chong-Min Kyung (S'76–M'81–SM'99–F'08) received the B.S. degree in electrical engineering from Seoul National University in 1975, and the M.S. and Ph.D. degrees in electrical engineering from KAIST, in 1977 and 1981, respectively. Since 1983, he has been with the KAIST, where he has been involved in CAD, computer graphics, microprocessors, DSP cores, and SoC's. In 2011, he founded and currently leads as the Director of the Center for Integrated Smart Sensors, a Global Frontier Project supported for nine years by the Korean Government. He has published over 300 papers in the international conferences and journals in the system-on-chip design, EDA algorithm, and system-level design. His current research interest is system-level optimization of smart sensors, especially 3-D smart cameras. He received the Best Paper/Design Awards in numerous international conferences, including ASP-DAC 1997 and 1998, DAC in 2000, ICSPAT in 1999, ICCD in 1999, and ISQED in 2014. He is a member of National Academy of Engineering Korea, and the Korean Academy of Science and Technology.