Using digital Phase-Locked Loop (PLL) technique for assessment of periodic body movement patterns on a mobile phone

Wac, Katarzyna; Paller, Gabor; Hausmann, Jody

How to cite

WAC, Katarzyna, PALLER, Gabor, HAUSMANN, Jody. Using digital Phase-Locked Loop (PLL) technique for assessment of periodic body movement patterns on a mobile phone. In: Workshop Proceedings of the 1st International Workshop on Frontiers in Activity Recognition using Pervasive Sensing in conjunction with Pervasive 2011 - IWFAR 2011. Plötz, T. & Roggen, D. & Bulling, A. (Ed.). San Francisco (United States). Newcastle : Newcastle University, 2011. p. 24–29.

This publication URL: https://archive-ouverte.unige.ch/unige:55850

Using digital Phase-Locked Loop (PLL) technique for assessment of periodic body movement patterns on a mobile phone

Gabor Paller
Sfonge
Hiilikatu 3
00180 Helsinki, Finland
gaborpaller@gmail.com

Jody Hausmann
University of Geneva
Geneva, Switzerland
jody.hausmann@unige.ch

Katarzyna Wac
University of Geneva
Geneva, Switzerland
katarzyna.wac@unige.ch

ABSTRACT

This paper applies digital PLL (Phase-locked loop) approach to the body movement classification problem. PLLs have been used efficiently in telecommunication to retrieve modulated signals from the background noise. Acceleration sensor signal processing often require the same kind of noise tolerance and the relevant movement patterns are frequently periodic, e.g. walking. The paper presents the PLL-based algorithm developed for step counting on a mobile phone and evaluates it against a commercially available wearable step counter.

Categories and Subject Descriptors

J.3 [**Life and Medical Sciences**]: Health

1. INTRODUCTION

A pedometer is "an instrument that gauges the approximate distance travelled on foot by registering the number of steps taken"[1]. Pedometers can be useful to estimate physical activity by counting steps and estimate walk and run speed. Metabolic Equivalent of Task (MET) is determined by walk speed and the duration of the activity and pedometers are critical components in such diverse applications as augmented reality [18] or aged care [7].

In the course of the last years, pedometers became very popular for personal wellness. Having the possibility to quantify the effort motivates a lot of people. Different studies show that pedometer users change their daily behavior to walk more steps per day [20], [21], [5]. There are many commercial pedometers available using different technologies. Some are mechanical and must be oriented; others detect steps in a 3D space. Specialized wearable motion detectors exist [22] but with the emergence of sensor-equipped smartphones and portable computers, installable pedometer applications became popular. Manufacturers also add a software layer on device to provide step count as well as different information like the quantity of calories burned. Many of these devices use proprietary formulas [20]. The accuracy of these devices also vary significantly. [8] found that the accuracy was variable and depended on the speed of the user; also quality depended a lot on the price. In [15], they compared the step values of 13 models and results showed that five models underestimated steps by 25% and three overestimated by 45%.

[1]The American Heritage Dictionary of the English Language

2. OVERVIEW OF THE STEP COUNTING ALGORITHMS

3-axial accelerometers became standard components of modern smartphones therefore applications acting on accelerometer data got into the focus of interest. Step counting is a well-known accelerometer application. In some cases the accelerometer's position relative to the body is known and fixed [6]. More frequently, however, the accelerometer is placed by the user, e.g. the user puts the device containing the accelerometer into his or her pocket or holds the device in hand. As our motivation was to use smartphone accelerometers to analyze body motion, we assume that the user fixes the accelerometer to his or her torso[2] (e.g. belt, pocket) and the accelerometer is not fixed to some hard to reach location like thigh or limb [9]. During the movement, different parts of the body perform characteristic motions and the torso is an ideal place to capture motion elements from the entire human body [22]. If the user does not swing his arms, holding the device in hand is equivalent to torso placement. The disadvantage of the torso placement is that the accelerometer's output signal is influenced by the acceleration caused by different body parts therefore more complex noise filtering has to be performed.

In case of user-placed accelerometer, the accelerometer's axes relative to the user's body are not known and no assumption can be made about the meaning of the individual axes. In this case the step counting algorithm can only take into account the absolute value (length) of the acceleration vector.

[2]Trunk or torso is an anatomical term for the central part of many animal bodies (including that of the human) from which extend the neck and limbs (Dorland's Medical Dictionary)

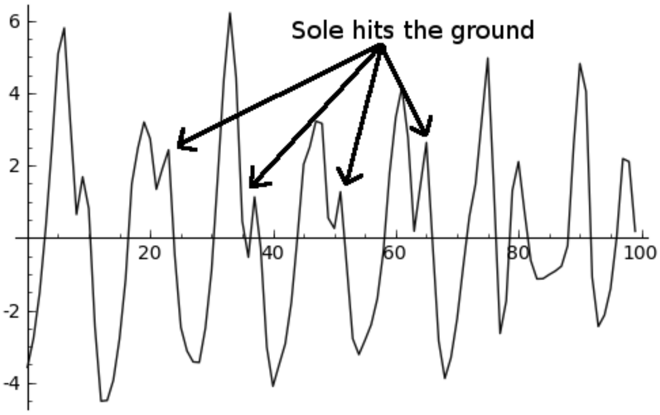

Figure 1: Typical acceleration signal caused by walking movement

$$a = \sqrt{x^2 + y^2 + z^2} \quad (1)$$

where x, y and z are acceleration values measured along the respective axis. Using Eq. 1, the step counting problem becomes a one-dimensional signal analysis problem.

Walking or running generates a periodic signal that superficially looks like a sine wave [19]. Whenever the center of the body moves upward, the acceleration caused by the body's movement is added to the Earth's gravity acceleration, causing positive peaks in the acceleration's absolute value. The opposite happens when the center of the body moves toward the Earth, then the body's acceleration is subtracted from the Earth's acceleration. An interesting effect can be observed when the sole hits the ground. The foot then dampens the impact causing a small peak in the acceleration's absolute value. If the gravity acceleration is subtracted, these effects create a wave signal with positive and negative periods. Figure 1 depicts a typical walking signal where x axis is the sample counter and y axis is the measured acceleration minus the Earth's gravity acceleration in m/s^2.

A number of algorithms have been proposed to detect walking movement. The most evident approach is to detect the walking signal in the time domain. These algorithms are most frequently based on peak detection with additional filtering of spurious noise spikes [11], [14], [1], [24]. Figure 6 demonstrates the limits of time-domain analysis. In this signal, the sole bounce-backs are non-typical which easily results in false peak detection and double-counting the steps. We implemented a naive step counter and tested on a similar signal where the test subject made 20 steps. The naive step counter first applied a lowpass 7-order IIR filter with 3 Hz pass-band limit frequency, then counted steps at each negative-positive transition of the filtered signal. The algorithm counted 34 steps. The false steps were due to sole bounce-back counted as steps in spite of the low-pass filtering and non step-related body movements counted as steps.

It is an attractive idea to detect the periodic walking signal in the frequency domain [2], [4]. This approach has limitations, however. It is relatively easy to detect the presence of the frequencies characteristic to walking (0.6-2.5 Hz) in the spectrum but extracting further characteristics, e.g. step count is more difficult to achieve. This is due to the fact that frequency-domain analysis provides one value about a set of samples in the frequency band and it is not trivial to deduct, if the user made 1 or 2 steps in the analysis window, particularly if the frequency resolution is low. Low frequency resolution may be a result of sampling frequency and analysis window length constraints. It is frequently the case that the analysis window has to be kept short in order to avoid that e.g. "walk" and "no walk" sections are present in the analysis window at the same time.

3. PHASE-LOCKED LOOPS

Phase-locked loops (PLL) are widely used in electrical engineering to derive a periodic signal from noisy input signal in real time. The assumption is that the input signal of the PLL is a periodic signal plus noise. The frequency of the internal oscillator of the PLL is then tuned so that the phase difference between the input signal and the signal of the internal oscillator is minimal. If the assumption about the periodic nature of the input signal holds, the internal oscillator's signal will be phase-locked to the input signal, thus can be considered a noise-free version of the input signal.

Key element of the PLL is the phase detector. The phase detector takes the input signal and the internal oscillator's signal and produces the phase difference which is in turn used to change the frequency of the internal oscillator. The phase detector may be analog or digital, depending on its input signal. Digital phase detector generates digital phase difference signal from digital input signals. If the internal oscillator and the input signal are analog, binary comparator is used to obtain the digital input signal for the phase detector.

4. USING PLL FOR STEP COUNTING

PLLs are particularly attractive for periodic acceleration signal processing because of their noise tolerance and because the output signal of their internal oscillator is easy to analyze as it is noise-free. Peak detection, phase calculation is easy to perform on the internal oscillator's output. Disadvantage of a PLL-based solution is the basic PLL assumption that there is a periodic input signal. If the PLL has no periodic input signal to lock into, its internal oscillator becomes free-running, generating a signal that has no relationship with the input signal. Lock detection is therefore a crucial issue.

Some properties of the walking signal also make it non-trivial to employ a PLL.

- The walking signal is not symmetric. The negative period often has smaller amplitude than the positive period and this property is more pronounced in case of vigorous movement, e.g. running or jumping. E.g. in Figure 6 the maximum amplitude of the positive phase is more than 8 but the maximum amplitude of the negative phase is about 4 (+0.8 g and -0.4 g relative to the Earth's acceleration of 1 g). Analog phase detectors detect false phase difference because the symmet-

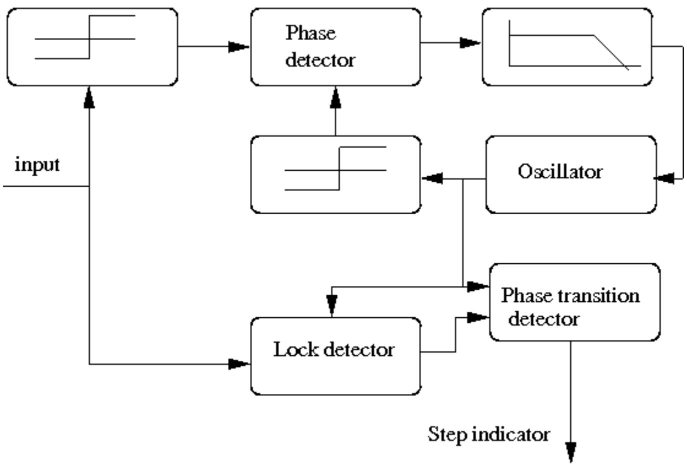

Figure 2: Block diagram of the step counting algorithm

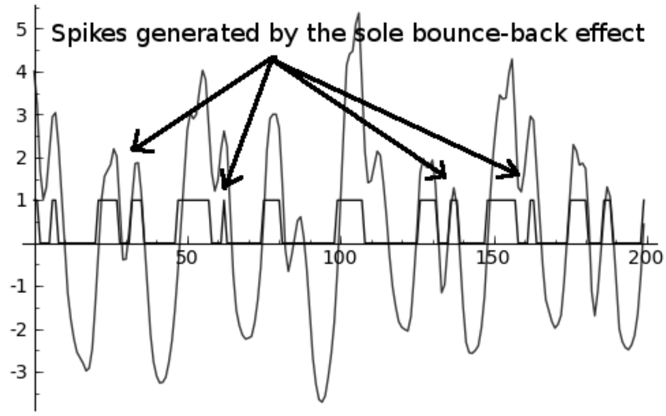

Figure 3: Motivation for spike filtering

ric sine wave the internal oscillator generates does not match the asymmetric input signal. For this reason, digital phase detectors are preferred.

- Walking may be a rhythmic movement but may also contain interruptions, single steps and no-walk sections. The PLL must lock into the signal very quickly else some steps will be missed. Also, there must be a reliable and fast lock detection mechanism to prevent step counting when the internal oscillator is free-running.

- Different kind of gait, ground and shoe sole types generate very different ground impact effect as demonstrated in Figure 1 and 6. In addition, if the device containing the accelerometer is not fixed to the user's body rigidly, acceleration noise may be added to the valuable signal when the accelerometer moves relative to the user's body. Unlike specialized wearable motion sensors, smartphones are very often not fixed rigidly to the user's body. The PLL may lock onto those false peaks and may count extra steps.

The design considerations above led to the PLL-based step counting algorithm depicted in Figure 2.

4.1 Binary threshold comparator and spike filter

The algorithm employs a digital phase detector. The reasons are the following. First, if the threshold of the input signal's comparator threshold is correctly set, the asymmetric nature of the input signal does not affect the digital signal fed into the phase detector. Second, the comparator eliminates smaller false peaks. This allows us to tune the PLL parameters for fast locking which would be impossible if the PLL had to cope with a lot of false peaks in the acceleration signal.

The comparator digitizing the input signal uses an adaptive threshold algorithm. This adaptive algorithm maintains the maximum and minimum values of the input signal and sets the threshold to slightly higher than the average of the maximum and the minimum value. Both the maximum and the minimum values are subject to exponential averaging with the last measured sample so that peaks are "forgotten" gradually. $\alpha = 0.05$ has been found a good compromise for the smoothing factor of exponential averaging. The algorithm is (x being the last measured sample):

```
if x>x_max then
    x_max=x
else
// exponential averaging with alpha=0.05
    x_max=x_max*0.95+0.05*x
if x<x_min then
    x_min=x
else
// exponential averaging with alpha=0.05
    x_min=x_min*0.95+0.05*x
threshold=(x_max+x_min)*1.05/2
```

The threshold of the internal oscillator's comparator is constant 0 as that signal is noise-free.

Spikes in the digitized input signal are filtered out before the input signal is fed into the phase detector. Some algorithms like the one presented in [11] use high threshold value to prevent the sole bounce-back effect to generate another false peak. This approach was not found to be satisfactory due to the high variation of peak values in certain persons' walking signal. If the threshold is set too high and the next step generates a significantly lower peak than the previous step, the step counter may miss a step. Instead we set the threshold slightly higher than the average of the maximum and minimum value and employ a spike filter. Figure 3 presents the absolute value of the analog input signal (Eq. 1) and the digitized input signal without spike filtering.

The spike filter employs a sliding window. If any value within the windows is '1', the output of the filter is also '1'. The default window length is 3 samples, determined experimentally. For certain persons with a particularly strong bounce-back effect it was necessary to increase the length of the filter to 6 samples.

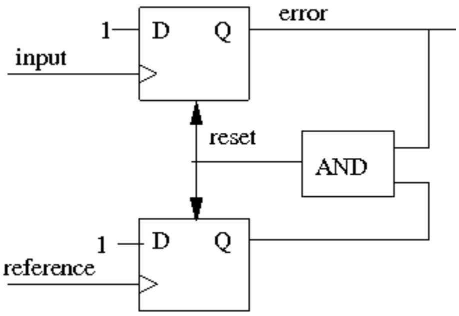

Figure 4: Phase detector arrangement used in the algorithm

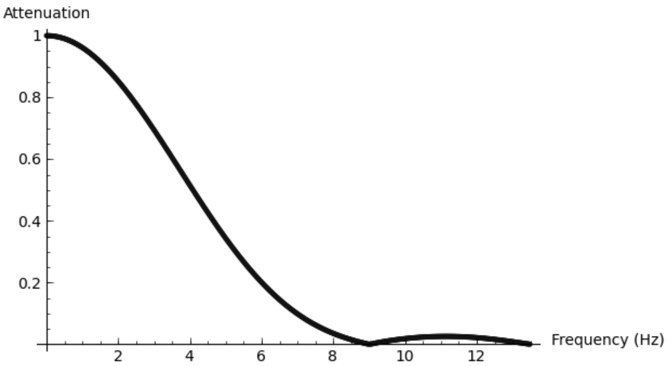

Figure 5: Loop filter frequency characteristics

4.2 Phase detector and PLL

There are a number of digital phase detector methods published in the literature, the simplest being a plain XOR operation that results in an error signal ("true" value) if its two inputs are not equal [3]. Even after the comparator, the digital input signal is sufficiently noisy so that this simple phase detector does not yield good enough result. A popular digital phase detector was adapted [3] that looks for the rising edge of its input signals (Figure 4). Whenever the "input" signal goes from 0->1, the phase detector generates a "1" error signal. This error signal is cleared when the "reference" signal makes a 0->1 transition. Used with the PLL feedback loop, the phase detector tries to enforce that a rising edge of the "input" signal is followed by a rising edge of the "reference" signal as soon as possible, yielding phase-locked reference signal.

The error signal is then fed into a 6-order low-pass FIR filter that acts as PLL loop filter. The relatively mild loop filter guarantees that the PLL locks quickly to the input signal (see Figure 5 for the filter characteristics). The output of the loop filter is used to change the internal oscillator's frequency. The base frequency of the oscillator is about 0.56 Hz (determined by the signal generated by the slowest walking),

its maximum frequency is about 6.7 Hz (determined by the need of the PLL to lock quickly to input signal). The wide frequency range allows the PLL to lock into vigorous movements like fast running.

4.3 Lock detection

Fast and reliable lock detection is critical as the PLL's internal oscillator is always running, even if there is no input signal. The literature proposes a wide range of lock detectors [10],[16],[17] but none of them was found reliable enough for our purposes. Our lock detector is based on the cross-correlation of the analog input signal and the internal oscillator's analog signal.

Cross-correlation between the analog input signal and the internal oscillator's signal is calculated as the following from the base sample of n:

$$c[n] = \sum_{k=0}^{N} a[k+n]y[N-k+n] \tag{2}$$

where $a[n]$ is the analog input signal, $y[n]$ is the output of the internal oscillator and $N = 10$ was used as convolution window. The convolution window has to be short enough so that there is only one cycle of the input/reference signal in the window while long enough so that the convolution signal reliably reflects the similarity of the input and the reference signal. The power of this signal is calculated for a single period of the internal oscillator's signal.

$$p_c = \sum_{k=n}^{n+t_c} c[k]^2 \tag{3}$$

where t_c is the cycle period of the internal oscillator during the time of the power value calculation. When the comparator of the internal oscillator detects a 0->1 transition (that is, the internal oscillator starts a new cycle), the following equation is evaluated.

$$p_a = \frac{p_c}{t_c(\max c[n] - \min c[n])} 10 \tag{4}$$

where $\max c[n]$ and $\min c[n]$ are the maximum and minimum values of Eq. 4 in the cycle. The resulting p_a value determines whether a step is detected or not at the end of the internal oscillator's cycle. We collected walking samples from 18 people and we found that limit value of 3.0 works well for most persons. In case of persons with less dynamic walking pattern, the limit value may need to be lowered.

Figure 6 demonstrates the operation of the algorithm on the "problematic" walking signal with strong sole bounce-back effect. The figure shows the original acceleration signal, the internal oscillator's output signal and the step detector's output (spike-like signal when a step was detected). It can be observed that the step detector correctly recognized the steps even in the presence of significant noise.

5. EXPERIMENTS AND RESULTS

The algorithm was developed using the open-source mathematical program called Sage [3]. In this phase, recorded

[3]http://www.sagemath.org

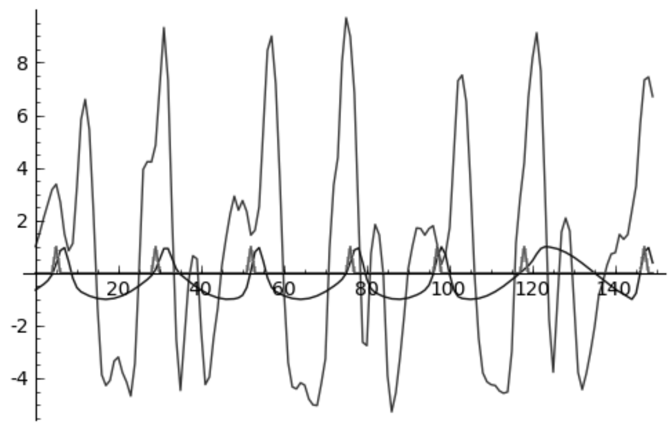

Figure 6: Demonstration of the step detection algorithm

accelerometer samples were used to experiment with the algorithm. Then a prototype was developed for Android mobile operation system. The test devices were Google Nexus One and HTC Desire phones with Android 2.2 release. These phones have built-in triaxial accelerometers (Bosch BMA150). In order to evaluate the feasibility of the PLL-based step detector idea, the maximum accelerator sampling frequency available on the device was used which is about 27 Hz. Note that the Android sensor API does not allow the specification of exact sampling frequency, only sampling frequency categories which may depend on the manufacturer. For example the DELAY_FASTEST category on Nexus One yields about 27 Hz sampling frequency while on Sony-Ericsson X10 Xperia Mini Pro, the sampling frequency is about 95 Hz in the same category. The prototype implementation does not detect the sampling frequency automatically and it expects a sampling frequency of about 27 Hz. The sampling frequency is not constant either, on the Nexus One it varies between 25-28 Hz.

When the step detection was in operation, a background service constantly sampled the accelerometer, ran the algorithm and if the user interface was visible, sent the result to the user interface for displaying the steps to the user. This continuous sampling decreases the battery life significantly because the smartphone's high-performance main CPU (1 GHz clock speed in case of Nexus One) has to be woken up from its idle state for gathering and processing each individual sample. Low-power auxiliary processor has been proposed to relieve the main processor from the background processing and allowing it to sleep [13] but this approach has not been adopted in commercially available devices.

We did not have the possibility to conduct a proper comprehensive testing of the prototype but we did make an effort to evaluate the prototype with 8 persons (2 females, 6 males). These persons walked with the smartphone in hand 50-100 steps making effort not to swing the phone in their hands and counted their steps. The floor was a normal office floor with no stairs and they had to change speed after every 10-20 steps as they had to go around corners. The shoes were different ranging from slippers to street shoes. As noted previously, if the user does not swing his or her hand, holding

Counted by the person	Our prototype
60	60
44	46
40	42
43	42
56	56
100	99
100	98
100	102

Table 1: Steps counted by the person and by our prototype

Counted by the person	Our prototype	SenseWear
100	102	96
100	108	101
200	225	210
100	102	99

Table 2: Parallel testing of BodyMedia SenseWear and our algorithm

the phone in hand is equivalent with attaching the phone to the trunk. Before the test, 3-4 steps were made to set the sensitivity suitable for the person. The results can be seen in Table 1. The average error is 2.1%. The error was calculated as $(steps_{measured} - steps_{counted})/steps_{counted}$ for every row. Error values per row were then averaged.

The prototype was also evaluated by testing it in parallel with BodyMedia SenseWear [4][23]. One healthy male test subject walked distances between 100-200 steps, counted the steps himself and read both step counters. The test was made outdoor, on different terrains that included concrete pavement and walking trails too and the tester wore outdoor shoes. It must be noted that SenseWear is an armband thus it is fixed to the user's body while the smartphone was held in the tester's hand and was subject to inadvertent arm movements even though effort was made not to swing the arm. The results can be seen in Table 2. The average error was 6.125% for our prototype and 2.75% for SenseWear. The average error was calculated in the same way as in the previous measurement.

6. CONCLUSIONS AND FUTURE WORK

Accelerometers in torso-mounted smartphones are able to provide data about movements involving the entire body. Extracting features from this composite signal may be challenging, however. In this paper we aimed to demonstrate that the well-known noise tolerance of phase-locked loops can be utilized to process accelerometer signals and extract periodic features, e.g. step count.

We found that the first results are promising and our prototype was comparable in precision to BodyMedia SenseWear though there is a space for further improvements. In particular, the sensitivity and spike filter length sometimes needs to be adjusted to the particular person whose steps are being counted. Settings these parameters adaptively would improve the user experience significantly. Decreasing the sam-

[4]http://sensewear.bodymedia.com/

pling frequency would also decrease the power consumption of the smartphone when the accelerometer is being sampled in the background.

It was found that even though smartphones can be used in human movement-related accelerometry, they have their limitations too. Smartphones on the market, particularly the high-end ones are not suitable for continuous sensor monitoring due to the increased power consumption of their main processors. Also, the lack of precise control and the variance of the sampling frequency of the Android platform on which the prorotype was developed complicates sensor processing implementation.

7. REFERENCES

[1] AHOLA, T. 2010. Pedometer for Running Activity Using Accelerometer Sensors on the Wrist, *Medical Equipment Insights 2010:3 1-8*

[2] BARALLON, P. , VUILLERME, N., AND NOURY, N. 2006. Walk Detection With a Kinematic Sensor: Frequency and Wavelet Comparison, *Proceedings of the 28th IEEE EMBS Annual International Conference, New York City, USA, Aug 30-Sept 3, 2006*

[3] BEST, R. 2007. Phase Locked Loops 6th edition: Design, Simulation, and Applications, *McGraw-Hill, 2007*

[4] BIDARGADDI, N., SARELA, A., KLINGBEIL, L. AND KARUNANITHI, M. 2007. Detecting walking activity in cardiac rehabilitation by using accelerometer, *3rd International Conference on Intelligent Sensors, Sensor Networks and Information, 2007. ISSNIP 2007, Brisbane, Australia*

[5] CONSOLVO, S., EVERITT, K., ET AL. 2006. Design requirements for technologies that encourage physical activity, *Proceedings of the SIGCHI conference on Human Factors in computing systems, Montreal, Quebec, Canada, 2006.*

[6] FOERSTER, F. AND FAHRENBERG, F. 2000. Motion pattern and posture: Correctly assessed by calibrated accelerometers, Behav. Res. Methods Instrum. Comput. 2000 Aug;32(3):450-7.

[7] GAO, J. AND KORONIOS, A. 2010. Mobile Application Development for Senior Citizens, *Pacific Asia Conference on Information Systems (PACIS) 2010*

[8] JAKICIC, J. ET AL. 2005. Measuring Energy Expenditure and Physical Activity Review of Literature, *http:// pmbcii.psy.cmu.edu/ core_c/ physical_activity.html, 2005*

[9] MAYAGOITIAA, R.E.,NENEB, A. V. AND VELTINK, P.H. 2002. Accelerometer and rate gyroscope measurement of kinematics: an inexpensive alternative to optical motion analysis systems, *Journal of Biomechanics 35 (2002) 537-542*

[10] MELIKYAN, V., HOVSEPYAN, A., ISHKANYAN, M. AND T. HAKOBYAN, T. 2008. Digital lock detector for PLL, *Design & Test Symposium (EWDTS), 2008 East-West, Lviv, 2008*

[11] MLADENOV, M. AND MOCK, M. 2009. A step counter service for Java-enabled devices using a built-in accelerometer, *CAMS '09, Proceedings of the 1st International Workshop on Context-Aware Middleware and Services: affiliated with the 4th International Conference on Communication System Software and Middleware (COMSWARE 2009)*

[12] PERROTT, M.H. 2009. Tutorial on Digital Phase-Locked Loops, *Custom Integrated Circuits Conference, Sept. 2009*

[13] PRIYANTHA, B.,LYMBEROPOULOS, D. AND LIU, J. 2010. Little Rock: Enabling Energy Efficient Continuous Sensing on Mobile Phones, *Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks, New York, USA, 2010*

[14] RANDELL, C., DJIALLIS, C. AND MULLER, H. 2003. Personal Position Measurement Using Dead Reckoning, *ISWC '03 Proceedings of the 7th IEEE International Symposium on Wearable Computers*

[15] SCHNEIDER, P., CROUTER, S., ET AL. 2004. Pedometer measures of free-living physical activity: comparison of 13 models, *Medicine & Science in Sports & Exercise 36(2): 331., 2004*

[16] STENSBY, J. 1992. Lock detection in phase-locked loops, *SIAM Journal on Applied Mathematics, Volume 52 Issue 5, Oct. 1992*

[17] STENSBY, J. 2005. An improved lock detector for phase-locked communication receivers, *Journal of the Franklin Institute, Volume 342, Issue 2, March 2005, Pages 149-159*

[18] TOKUSHO, Y. AND FEINER, S. 2009. Prototyping an Outdoor Mobile Augmented Reality Street View Application, *8th IEEE International Symposium on Mixed and Augmented Reality, 2009. ISMAR 2009*

[19] TORREALBA, R.R, CASTELLANO, J.M. , FERNANDEZ-LOPEZ, G., AND GRIECO, J.C. 2007. Characterisation of gait cycle from accelerometer data, *Electronics Letters, September 27, 2007*

[20] TUDOR-LOCKE, C. 2002. Taking Steps Toward Increased Physical Activity: Using Pedometers to Measure and Motivate, *PresidentâĂŹs Council on Physical Fitness and Sports, Research Digest, June 2002*

[21] TUDOR-LOCKE, C., BASSETT, D. R. ET AL. 2004. A preliminary study of one year of pedometer self-monitoring, *Annals of Behavioral Medicine 28(3): 158-162.*

[22] YANG, C.-C. AND HSU, Y.-L. 2010. A Review of Accelerometry-Based Wearable Motion Detectors for Physical Activity Monitoring, *MDPI Sensors 2010, 10, 7772-7788*

[23] WELK, G. J., ET AL. Field Validation of the MTI Actigraph and BodyMedia Armband Monitor Using the IDEEA Monitor. *Obesity, 15, 4, (2007), 918-928.*

[24] ZHONG, S., WANG, L., BERNARDOS, A. M. AND SONG, M., 2010. An Accurate and Adaptive Pedometer Integrated in Mobile Health Application, *IET International Conference on Wireless Sensor Network (IET-WSN 2010), July 2010, Beijing, China*