

An Automatic Segmentation Technique in Body Sensor Networks based on Signal Energy

Eric Guenterberg, Sarah Ostadabbas, Hassan Ghasemzadeh, Roozbeh Jafari
Department of Electrical Engineering, University of Texas at Dallas
Richardson, TX 75080-3021

{mavpion, sarahostad, h.ghasemzadeh}@student.utdallas.edu, rjafari@utdallas.edu

ABSTRACT

Monitoring human activities using wearable wireless sensor nodes has the potential to enable many useful applications for everyday situations. The long-term lifestyle monitoring can greatly improve healthcare by gathering information about quality of life; aiding the diagnosis and tracking of certain diseases such as Parkinson's. The deployment of an automatic and computationally-efficient algorithm reduces the complexities involved in the detection and recognition of human activities in a distributed system. This paper presents a new algorithm for automatic segmentation of routine human activities. The proposed algorithm can distinguish between discrete periods of activity and rest without specifically knowing the activity. A finite subset of nodes can detect all human activities, but each node by itself can only detect a particular set of activities. For local segmentation we choose the parameters for each node that result in the least segmentation error. We demonstrate the effectiveness of our algorithm on data collected from body sensor networks for a scenario simulating a set of daily activities.

Keywords

Automatic Segmentation, Body Sensor Networks, Physical Movement Monitoring.

1. INTRODUCTION

The development of small sensor platforms from relatively inexpensive, commercially available components has created fascinating new opportunities for data collection. The sensor nodes can communicate wirelessly, have limited storage capabilities, and are often deployed in networks. These sensor networks have been used for a wide variety of sensing tasks from individual health monitoring to large-scale environmental sensing. The sensors measure environmental variables including

temperature, humidity, force, acceleration, and heartbeat. In most applications, a light-weight embedded sensor node is expected to acquire physical measurements, perform local processing and storage, and communicate over a short distance. Sensor networks that understand human actions are expected to be useful for numerous aspects of everyday life. Human motion recognition using wireless sensor networks can employ data from either environmental or on-body sensing devices [1]. One method of using environmental sensors is video processing. The use of video streams in wireless networks to interpret human motion has been considered for assisted living applications [2]. In contrast, wearable sensor networks or body sensor networks (BSNs) are built of light-weight sensor platforms and can be used to recognize the actions performed by the person wearing them [3]. The sensors can be mounted on the human body or even woven into the clothing itself [4]. Unlike vision-based platforms, BSNs require no environmental infrastructures and they are less expensive. Moreover, the signal readings from on-body sensors are clearer and not biased by environmental effects such as light and the background. This makes BSNs potentially more accurate than vision frameworks for motion recognition.

For clinical systems, BSNs offer the unprecedented ability to monitor patients in a natural setting for an extended period. For this application, data reduction is critical: a clinician or doctor cannot examine reams of data to assess the patient's performance and health. Instead, the doctor might prefer a summary, such as "the patient ran three miles on Tuesday, but slept all Wednesday. She has poor balance in the morning, but after three o'clock her balance improves considerably. She fell 10 times last week." This example suggests that the system should employ pattern recognition techniques to classify the data into a set of actions. The BSNs in these settings must operate for extended periods of time without communicating with a base station, therefore the signal processing and classification must be performed *in situ*. The limited storage and processing capabilities provide many challenges to architects of these systems.

Pattern recognition typically involves four steps. First, the data from the sensors is sent through a series of filters to reduce noise, apply calibration to the data, etc. Subsequently, the data is partitioned into individual segments to be classified. An example is that an image of a page of text can be segmented into words or even letters without knowledge of which letters or words are present. After this, features are extracted from each segment. Features are chosen to reduce the volume of data without

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

BodyNet'09, April 1-3, 2009, Los Angeles, USA.

Copyright 2008 ICST 978-963-9799-41-7.

eliminating relevant information. For instance, to recognize fruit, the color, shape, and size are relevant features, while the rotation and individual pixel values are irrelevant. Finally, the features are used to classify the segments. These classes are labels which range from individual actions such as “falling,” “sitting” or “walking” to conclusions about health, such as “erratic heartbeat.”

While image segmentation has been a much-researched topic, relatively little work has been done for segmentation of stream signals. The most popular approaches include fixed time slices and manual segmentation. Manual segmentation is acceptable for a twenty minute treadmill activity, but for a sensor network deployed for hours, days, or even weeks, automatic segmentation is required. Regions of interest in the data stream can be varying lengths, so a fixed time slice could contain part of an action, or potentially several actions. Certain classifiers, such as the Hidden Markov Model (HMM) can utilize fixed time slices effectively [5], especially when computationally expensive operations such as eigenvector extraction and Fourier transforms are used. However other classifiers popular for their low processing overhead such as k-NN and Adaboost cannot. Therefore, to enable use of the widest variety of classifiers, automatic classification capable of accurately segmenting sensor readings into discrete actions and non-actions is necessary.

In this paper we introduce a segmentation technique based on the notion that regions of interest in a data stream correspond to times when the readings from a sensor change rapidly during a short period of time. A “rest” is defined as a time during which the sensor’s values remain relatively constant over an interval. Empirically this is true for even seemingly continuous actions, such as walking. We present a framework for automatically segmenting actions in a sensor stream. This framework takes advantage of correlation in the observation of an action between sensor nodes and sensors to increase accuracy. A new adaptive method of determining activity in a data stream is explored. It is based on standard deviation of sensor data. Finally, we experimentally verify these techniques by comparing the results to our manual segmentation.

2. RELATED WORK

BSNs with inertial sensors have been used in several applications such as fall detection, gait analysis, sport medicine and balance assessment, and have received much attention recently. Many recent wearable systems for activity recognition place a single type of sensor, typically accelerometers, in multiple locations (anywhere from two to 12) on the body [6, 7]. More recent systems prefer using multiple motion sensors [8-10]. In order to provide relevant information to mobile users, [9] focuses on the recognition of activities that are characterized by a hand motion and an accompanying sound using on-body sensing. Jafari et al. [10] propose a wearable movement monitoring platform. The system consists of lightweight wireless sensor nodes, each equipped with accelerometers and gyroscopes. Sensor readings are separated into primitive actions by manual segmentation in an offline manner. At the next stage, they are classified using a k-NN classifier. [11] introduces a system for activity monitoring using a clustering approach and hierarchical framework. They utilize unsupervised learning to categorize a dataset and subsequently construct a hierarchy of relationships between clusters. They

segment the primary recordings by dividing the signal into fixed time slices with a 66% overlap. Renevey et al. [12] present a technique for activity classification which is used to improve the heart rate estimation in a system composed of optical probes and accelerometers. Feature vectors are extracted from autocorrelation matrices based on data from fixed size segments. In [13] and [14], Chambers et al. present classification results of human gesture for video annotation and retrieval. To perform segmentation, they examine stationary portions of the signal by employing a sliding window. The authors assume that the acceleration magnitude is very close to the magnitude of gravity for stationary portions; hence they assume a model of gravity plus Gaussian white noise. They measure the difference between log likelihoods of two adjacent windows. A sharp change in the log likelihood corresponds to the commencement of a new action/rest. A gesture recognition platform composed of wireless inertial measurement units is proposed in [15]. The variance of the data over a fixed window is measured as the potential metric for activity detection. Periods of activity are determined where the variance is greater than a constant threshold value.

Although all aforementioned techniques have been successful in providing a system of physical action classification, they may either use a manual segmentation approach to map original sensor readings to primitive actions (e.g. [10]) or utilize a special purpose auto-segmentation scheme. For instance, the fixed time slice techniques used in [11] and [12] provide excellent results while avoiding segmentation completely for certain classification schemes; but are not appropriate for others. We are investigating more general segmentation techniques that are capable of integrating with a wide range of classifiers. The technique in [13] and [14] does capture entire movements that occur over differing intervals, but it depends on specific sensor types and a relatively specific posture. The segmentation scheme in [15] uses a hard threshold value and do not take the advantage of activity correlation between data streams and sensor nodes.

Another platform for the classification of human movements is introduced by Mathie et al. [16]. Although they can reliably and easily distinguish between periods of activity and rest, the measurement device they use is selected very specifically. A similarity measure for segmenting and classifying motion streams is presented in [17]. Motion segments are generated by collecting data from primary stream and comparing with predefined reference motions by using some similarity measures.

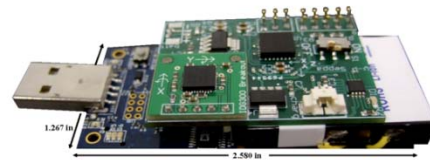


Figure 1. Inertial sensor board attached to a mote

3. SYSTEM ARCHITECTURE

We are using BSNs for physical movement monitoring. Each sensor node, also called a mote, is equipped with a custom-

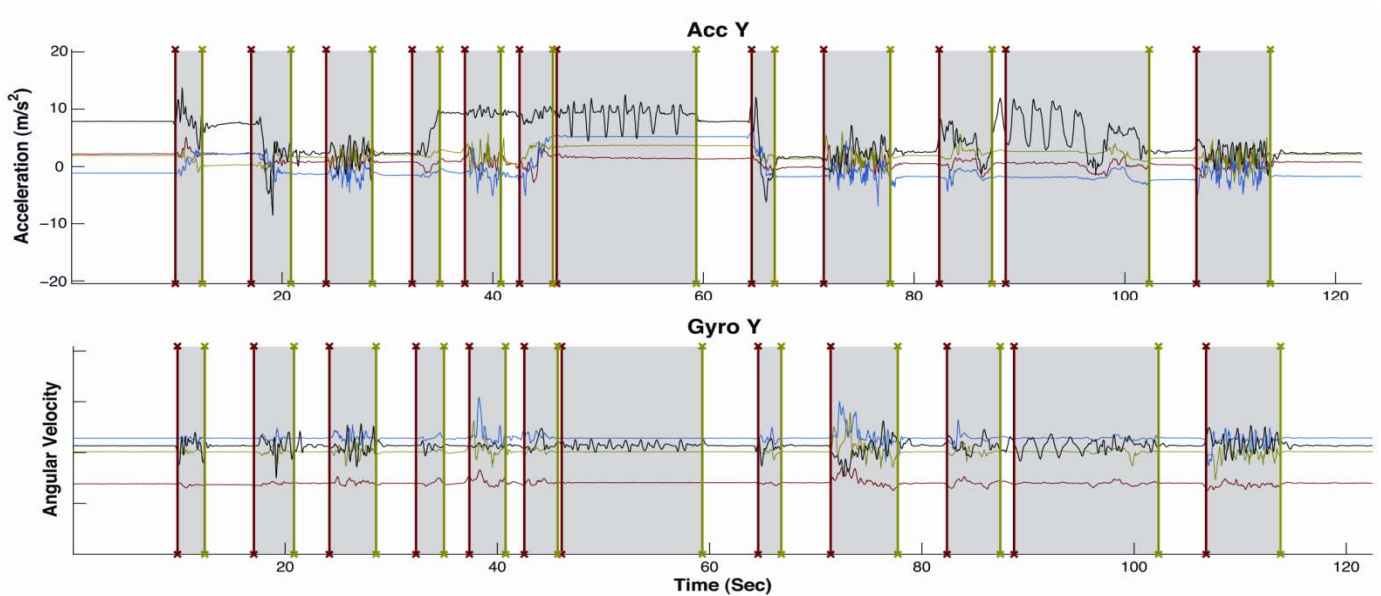


Figure 3. Manual Segmentation: white is rest, gray is action

designed sensor board with several inertial sensors as shown in Figure 1. We use the TelosB mote, which is commercially available from XBow®. The mote has a microcontroller for processing and storage and a radio for communication. Our custom-designed sensor board features a tri-axial accelerometer and a bi-axial gyroscope. The mote and sensor board are powered by a Li-ion battery integrated with each node. The motes sample their sensors at 50 Hz and use a TDMA scheme to communicate all data to an off-body base station. This sampling rate is experimentally chosen to provide sufficient resolution while compensating for the bandwidth constraints of our sensor platform. The base station relays the information to a PC via USB for post-processing.



Figure 2. Node placement of experimental subject

We collect and synchronize the sensor and video data in MATLAB and perform further analysis off-line. Our system consists of four sensor nodes arrayed on the body as shown in Figure 2.

4. MANUAL SEGMENTATION

Manual segmentation can be performed using one of several approaches. Logging the time of each action is a form of segmentation. Signals can be segmented by synchronizing a video to the signals and dividing them based on actions observed on screen. It is also possible to segment into actions by directly examining the signals and utilizing knowledge of the experimental technique.

Table 1. The scenario of daily activities

Action (initial Position : Lying)
1. Lie to sit
2. Sit to stand
3. Walk to a desk
4. Grasp a plate
5. Walk to a dining table
6. Sit and leave the plate on the table
7. Start eating
8. Sit to stand
9. Walk to a fridge
10. Pick a glass from the fridge
11. Drink and leave the glass on the fridge
12. Walk to the bed

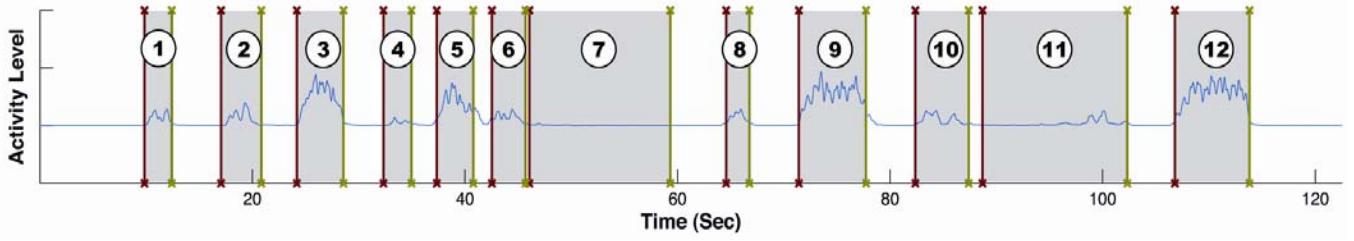


Figure 6. Activity level signal for the right wrist

Figure 3 shows a portion of the sensor signals from the four sensor nodes for the daily activities and highlights the segmentation choices. Each plot in the diagram contains the signals from all the sensor nodes for a given sensor type (such Y acceleration and rotation around Y). Table 1 shows 12 different movements present in our scenario. The segments can be distinguished by looking at the video and at the rapid changes in the sensor values.

We use manual segmentation as a ground truth with which to compare the results from segmentation. Where the automatic segmentation differs, it is considered to be in error.

5. AUTOMATIC SEGMENTATION

In the signal processing framework presented here, segmentation and classification are completely independent. Therefore for segmentation, the morphology of the signal for specific actions cannot be utilized, as it is not yet known. Furthermore, as it is important to limit inter-node communication, communicating the complete sensor readings between nodes is undesirable. We introduce a pure signal processing approach to this problem. Figure 4 shows the block diagram of our proposed method. It consists of three parts. First we calculate the activity level signal for each node and then put an adaptive threshold on it to compute the segmented signal which shows the activity vs. rest. At the end we remove the overly short rests and actions by a transient elimination technique.

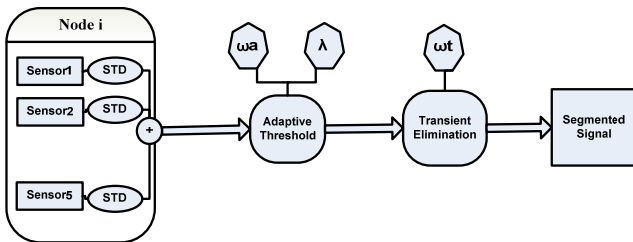


Figure 4. The block diagram of automatic segmentation

5.1 Per-sensor Activity Level

Each sensor data stream is filtered to provide a signal representing the level of activity at any given time.

The “range” of change in a data stream for a short time interval is represented by the standard deviation over the interval:

$$\sigma_n^2(x, i) = \frac{\sum_{j=i-(n-1)/2}^{i+(n-1)/2} (x_j - \mu_n(x, i))^2}{n} \quad (1)$$

where:

$$\mu_n(x, i) = \frac{\sum_{j=i-(n-1)/2}^{i+(n-1)/2} x_j}{n} \quad (2)$$

The window is of size n and is centered on point x_i .

5.1.1 Normalization of Activity level

The human body is an imprecise machine, therefore even when movement occurs primarily on one axis; some activity is typically measured on every axis. The activity level is correlated so adding the activity levels observed by each sensor will raise Signal-to-Noise Ratio (SNR), and improve the final segmentation. The maximum and minimum activity levels could vary significantly between sensors. Normalizing the activity levels will prevent one sensor from overly influencing the segmentation. This gain normalization should be actively controlled by the surrounding activity level to create a consistent normalization for more and less energetic actions. This leads us to normalize the activity level signal by dividing it by the mean of recent activity levels. This is shown in Figure 5.

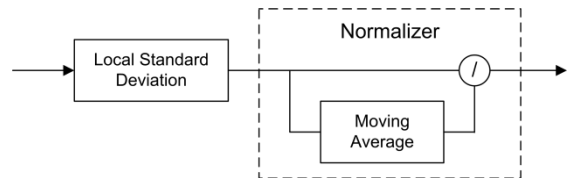


Figure 5. Standard deviation approach

5.1.2 Per-node Activity level

We calculate the standard deviation of each sensor (a tri-axial accelerometer and a bi-axial gyroscope) and add them up for each node. The result is a per-node activity level signal. Figure 6 shows the activity level signal from one of the sensor nodes for the daily movement, as described in Table 1. Corresponding manual segmentation has been highlighted on this figure to show that for

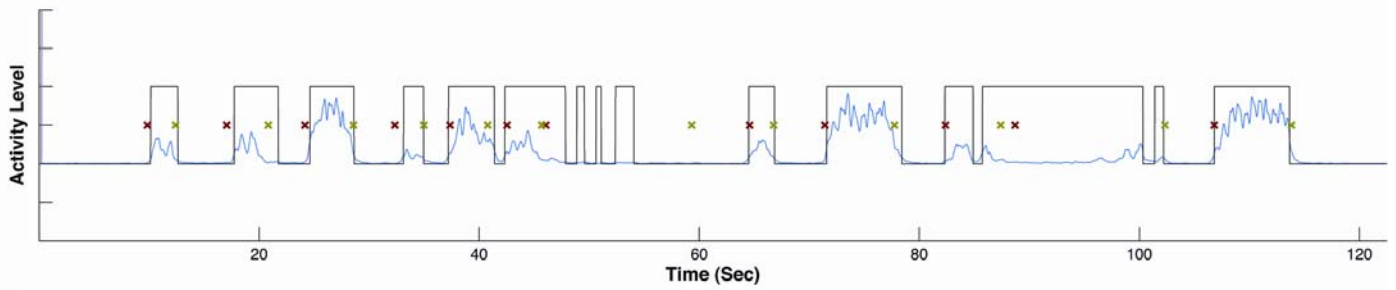


Figure 7. Segmented signal for node on right wrist

a number of activities just by looking at the raw signal can detect the start and stop. But for some of movements such as movement 7 and 11, it is difficult to detect the start and stop of the movement.

5.2 Adaptive Threshold

The activity level signal shows more changes and energy during activities. We use this property for our automatic segmentation. In human daily activities some actions have more energy than others, for example you can compare eating (movement 7) and walking (movement 9) in Figure 6. Therefore, a fixed threshold will fail to detect all activities. We propose an adaptive threshold method for solving this problem. This method is also a standard deviation-based approach. We calculate the standard deviation of the activity level signal in a window size, ω_a , and multiply by λ , then compare the activity level signal with this adaptive threshold to segment the signal. If the activity level is above the threshold, it is considered as an activity, if it is below, it is considered as rest.

5.3 Transient Elimination

Valid rests and activities have a minimum duration. We use this fact to remove the steps with duration smaller than a transient time interval, ω_t . It means that if the duration of activity in activity level signal is less than ω_t , we consider it as rest and vice versa.

As the diagram of Figure 4 shows there are three tunable parameters that can be adjusted for each node. The effects of changing each parameter are studied in the next section

6. EXPERIMENTS

In this section, we present our experimental validation of this segmentation technique by applying it to experimented data representing several daily activities.

6.1 Experimental Procedure

For the experiments, four right-hand subjects performed twelve movements that are shown in Table 1. The initial position and final are lying on the bed with about 2 minutes of activities in between. Each subject wore the sensor nodes described above in

positions shown in Figure 2. They were placed on the right wrist, one on the waist and one on each leg.

The results were segmented in two ways: manually and using the localized standard deviation of sensor activity. Manual segmentation is used as the ground truth. Errors in the automatic tools are anything that differs from manual segmentation. As each node could detect a particular set of activity not all of them, the error has been defined only on detected movements for every node.

6.2 Results

The data collected by our base station were filtered first to remove the effect of noise. Manual segmentation was performed with the help of the video recorded during data collection. Activity level signals were extracted by calculating standard deviation on a moving window of 10 samples over the set of sensor data for each node. For the adaptive threshold, we have changed the adaptive window size, ω_a , from 20 to 80 samples and the constant factor, λ , took values of 1, 2, 3 and 4. We considered the effect of the transient time interval, ω_t , by changing it from 10 to 40 samples.

Tables 2-5 show the segmentation error corresponding to variation of the above parameters for each node. The best choice for each has been highlighted. All nodes except right wrist missed some movements in detection. The node on the waist couldn't detect movements 7 and 11, while the two legs missed the movements 4, 6, 7 and 11 (see Table 1 for the movement names). The error values are averaged over the detected movements. These values show the difference between the manual and automatic segmentation on detecting the start and stop of an activity that a particular node could detect. Figure 7 shows the segmented signal for the right wrist node. The stars show the start and stop of the manual segmentation. From Figure 7, it is obvious that the segmentation is accurate for many movements, but for others the error is significant. Some of the errors are from transients that had been too big for the transient elimination part, suggesting that the window size of that should be tuned more aggressively. Most of the other errors resulted from separating one movement into several movements. For instance, in movement 7: "eating" sometimes the subject performed the movement fluidly, so it counted as one movement, and sometimes the subject paused, causing the movement to be recognized as several actions. The only reason manual segmentation recognized it as one action is because we *knew* it should be one action. If these types of errors are removed from the list, the overall error rate would be less than the values shown in the Tables 2-5.

The sampling frequency is 50Hz, so every sample is equal to 20 msec. We've shown the average errors in time scale and the average error percentage of detecting the start and stop of each movement in Tables 2-5, then Table 2 shows the best choice for parameters will result an average error of 9.1% in detecting a particular movement for right wrist. Given that features extracted from segmented movements are not significantly affected by small deviations of segment start and stop time, the error due to segmentation is not likely to cause serious classification error.

Table 2. The error correspond to changing the tunable parameters for right wrist

ω_a	λ	ω_t	Avg. Error	Avg. %
20	3	40	1.19 sec.	9.6%
40	3	40	1.23 sec.	9.7%
80	3	40	2.12 sec.	9.4%
40	2	40	1.18 sec.	9.1%
40	4	40	2.00 sec.	9.2%
40	1	40	1.48 sec.	11.9%
40	3	20	1.42 sec.	11.8%
40	3	10	1.50 sec.	11.9%

Studying the effects of changing parameters on our method's efficiency shows that by increasing ω_a each movement can be distinguished separately but it will cause a delay in detecting start and stop of each movement. λ controls the sensitivity of the adaptive threshold. A larger λ will result decrease detection of small movements. For transient elimination, the assumption that human body can switch from activity to rest or vice versa in time interval larger than ω_t lead to failure to segment some small movements between main activities.

For real-time applications, a delay is introduced equal to half of each window size. On the other hands increasing the window size, will increase computational complexity in each sample time. Thus there is a tradeoff between these tunable parameters.

Table 3. The error correspond to changing the tunable parameters for waist

ω_a	λ	ω_t	Avg. Error	Avg. %
20	3	40	0.77 sec.	6.4%
40	3	40	1.04 sec.	8.3%
80	3	40	1.78 sec.	10.4%
40	2	40	0.76 sec.	6.4%
40	4	40	1.44 sec.	11.5%
40	1	40	0.80 sec.	6.4%
40	3	20	1.24 sec.	9.9%
40	3	10	1.39 sec.	10.7%

Table 4. The error correspond to changing the tunable parameters for right leg

ω_a	λ	ω_t	Avg. Error	Avg. %
20	3	40	2.48 sec.	17.5%
40	3	40	3.24 sec.	19%
80	3	40	3.74 sec.	19.6%
40	2	40	2.12 sec.	17.3%
40	4	40	4.00 sec.	20%
40	1	40	1.54 sec.	12.1%
40	3	20	3.40 sec.	19.5%
40	3	10	3.45 sec.	19.6%

Table 5. The error correspond to changing the tunable parameters for left leg

ω_a	λ	ω_t	Avg. Error	Avg. %
20	3	40	2.34 sec.	17.1%
40	3	40	3.00 sec.	18.1%
80	3	40	3.86 sec.	19.2%
40	2	40	2.38 sec.	17.1%
40	4	40	1.92 sec.	16.9%
40	1	40	1.77 sec.	13.2%
40	3	20	3.28 sec.	18.2%
40	3	10	3.34 sec.	18.3%

7. CONCLUSION AND FUTURE WORK

In this paper we presented a signal processing model for segmenting data streams from inertial sensors into periods of activity and rest. This segmentation scheme is based on the idea that regions of interest in the signal involve rapid changes the data collected by the various sensors. This scheme utilizes the results from multiple sensors and nodes to increase accuracy. This segmentation is efficient enough to be deployed on sensor nodes. Our experimental evaluation highlights both the strengths and weaknesses of this technique. Most of the segmentation errors involved a single action being segmented as several actions. This was generally because a minority of nodes considered it one action, while the majority considered it two actions. By setting a voting threshold lower than 50% we should be able to decrease errors. Additionally, we should test segmentation on subjects in a less controlled setting with video to verify the effectiveness of this approach in more natural environments.

Acknowledgments

The authors would like to thank Ville-Pekka Seppa for his efforts in designing the sensor board.

8. REFERENCES

- [1] H. Ghasemzadeh, J. Barnes, E. Guenterberg, and R. Jafari, "A Phonological Expression for Physical Movement Monitoring in Body Sensor Networks," in *The Fifth IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS)*, Atlanta, GA, 2008.
- [2] M. Philipose, K. P. Fishkin, M. Perkowitz, D. J. Patterson, D. Fox, H. Kautz, and D. Hahnel, "Inferring activities from interactions with objects," *Pervasive Computing, IEEE*, vol. 3, pp. 50-57, 2004.
- [3] A. Vehkaoja, M. Zakrzewski, J. Lekkala, S. Iyengar, R. Bajcsy, S. Glaser, S. Sastry, and R. Jafari, "A resource optimized physical movement monitoring scheme for environmental and on-body sensor networks," in *Proceedings of the 1st ACM SIGMOBILE international workshop on Systems and networking support for healthcare and assisted living environments* San Juan, Puerto Rico, pp. 64-66, 2007.
- [4] R. Jafari, F. Dabiri, P. Brisk, and M. Sarrafzadeh, "Adaptive and fault tolerant medical vest for life-critical medical monitoring," in *Proceedings of the 2005 ACM symposium on Applied computing* Santa Fe, New Mexico: ACM, 2005.
- [5] L. Rabiner, "A tutorial on Hidden Markov Models and selected applications in speech recognition," in *Proceedings of the IEEE*, pp. 77-86, 1989.
- [6] N. Kern, B. Schiele, and A. Schmidt, "Multi-sensor Activity Context Detection for Wearable Computing," *Proc. 1st European Symp. Ambient Intelligence (Eusai 03)*, LNCS 2875, Springer, pp. 220-232, 2003.
- [7] L. Bao and S. Intille, "Activity Recognition from User Annotated Acceleration Data," *Proc. 2nd Int'l Conf. Pervasive Computing (Pervasive 04)*, LNCS 3001, Springer, pp. 1-17, 2004.
- [8] R. Aylward and J. Paradiso. A compact, high-speed, wearable sensor network for biomotion capture and interactive media. In *IPSN*, 2007.
- [9] Ward, J.A. Lukowicz, P. Troster, G. Starner, T.E. , "Activity Recognition of Assembly Tasks Using Body-Worn Microphones and Accelerometers," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1553-1567, 2006.
- [10] R. Jafari, W. Li, R. Bajcsy, S. Glaser, S. Sastry, Physical Activity Monitoring for Assisted Living at Home, In *Proceedings of International Workshop on Wearable and Implantable Body Sensor Networks (BSN)*, March 2007, Aachen, Germany.
- [11] D. M. Sherrill, M. L. Moy, J. J. Reilly, P. Bonato, Using hierarchical clustering methods to classify motor activities of COPD patients from wearable sensor data, In *Journal of NeuroEngineering and Rehabilitation*, pp. 1-14, June 2005.
- [12] P. Renevey, R. Vetter, P. Celka, J. Krauss, Activity Classification Using HMM For Improvement of Wrist Located Pulse Detection, In *Proceedings of the Biosignal*, 2002.
- [13] G. S. Chambers, S. Venkatesh, G. A.W. West, H. H. Bui, Segmentation of Intentional Human Gestures for Sports Video Annotation, In *Proceedings of the 10th IEEE International Multimedia Modeling Conference (MMM04)*, 2004.
- [14] G. S. Chambers, S. Venkatesh, G. A.W. West, Automatic Labeling of Sports Video Using Umpire Gesture Recognition, Lecture Notes in *Computer Science, Structural, Syntactic, and Statistical Pattern Recognition*, pp. 859-867, 2004.
- [15] A. Y. Benbasat, J. A. Paradiso. An inertial measurement framework for gesture recognition and applications. In *Gesture and Sign Language in Human Computer Interaction*, volume 2298 of LNCS, pp. 9-20. Springer-Verlag, 2002.
- [16] M. J. Mathie, B. G. Celler, N. H. Lovell, A. C. F. Coster, Classification of basic daily movements using a triaxial accelerometer, In *Journal of Medical and Biological Engineering and Computing*, pp. 679-687, Springer Berlin/Heidelberg , September, 2004.
- [17] C. Li, S. Q. Zheng, B. Prabhakaran, Segmentation and Recognition of Motion Streams by Similarity Search, *ACM Transactions on Multimedia Computing, Communications and Applications*, Vol. 03, No. 03, pp. 1-26, 2007.