

# DexterNet: An Open Platform for Heterogeneous Body Sensor Networks and Its Applications \*

Philip Kuryloski <sup>◊,†,◦</sup>, Annarita Giani <sup>†</sup>, Roberta Giannantonio <sup>△</sup>, Katherine Gilani <sup>\*</sup>,  
Raffaele Gravina <sup>◦</sup>, Ville-Pekka Seppä <sup>□</sup>, Edmund Seto <sup>▽</sup>, Victor Shia <sup>†</sup>, Curtis Wang <sup>†</sup>,  
Posu Yan <sup>†</sup>, Allen Y. Yang <sup>†</sup>, Jari Hyttinen <sup>□</sup>, Shankar Sastry <sup>†</sup>, Stephen Wicker <sup>◊</sup>, Ruzena Bajcsy <sup>†</sup>

<sup>†</sup> Department of EECS, University of California, Berkeley, CA 94720

<sup>◊</sup> Department of ECE, Cornell University, Ithaca, NY 14853

<sup>▽</sup> School of Public Health, University of California, Berkeley, CA 94720

<sup>△</sup> Telecom Italia, Turin, Italy

<sup>◦</sup> WSN Lab sponsored by Pirelli and Telecom Italia, Berkeley, CA 94704

<sup>□</sup> Department of Biomedical Engineering, Tampere University of Technology, Tampere, Finland

<sup>\*</sup> Department of EE, University of Texas at Dallas, TX 75080.

## Abstract

We present an open-source platform for wireless body sensor networks called *DexterNet*. The system supports real-time, persistent human monitoring in both indoor and outdoor environments. The platform utilizes a three-layer architecture to control heterogeneous body sensors. The first layer called the body sensor layer (BSL) deals with design of heterogeneous body sensors and their instrumentation on the body. At the second layer called the personal network layer (PNL), the body sensors on a single subject communicate with a mobile base station, which supports Linux OS and the IEEE 802.15.4 protocol. The BSL and PNL functions are abstracted and implemented as an open-source software library, called *Signal Processing In Node Environment (SPINE)*. A *DexterNet* network is scalable, and can be reconfigured on-the-fly via *SPINE*. At the third layer called the global network layer (GNL), multiple PNLs communicate with a remote Internet server to permanently log the sensor data and support higher-level applications. We demonstrate the versatility of the *DexterNet* platform via several real-world applications.

## 1 Introduction

Wireless body sensor networks (BSNs) have been an emerging research area in the past five years. The develop-

ment is mainly due to two reasons: 1. Continuing integration and miniaturization of sensors, processors, and radio devices. 2. Rising demands for advanced body sensor systems ranging from pivotal areas of elderly protection and clinical patient monitoring to much broader applications in military, preventive healthcare, and consumer electronics. Traditional BSNs mainly involve single wearable sensors, such as fall detection [4, 14, 16], walk and gait-phase detection [1, 12], and pulse-oximetry monitoring [10, 11]. More sophisticated systems consist of multiple heterogeneous sensors, adopt certain hierarchical architectures for real-time sensor management, and may integrate body sensors with other environmental sensors. Existing systems include CodeBlue [8], HealthGear [11], MobiCare [3], WW-BAN [9], ALARM-NET [17], Participatory Sensing [2], and Intel MSP [6], to name a few. These systems instrument the human body as an active mobile platform, and have the necessary mobility to support persistent monitoring in people's normal living environments.

In this paper, we present a novel platform for heterogeneous body sensor networks called *DexterNet*. The design principles of *DexterNet* are manifold:

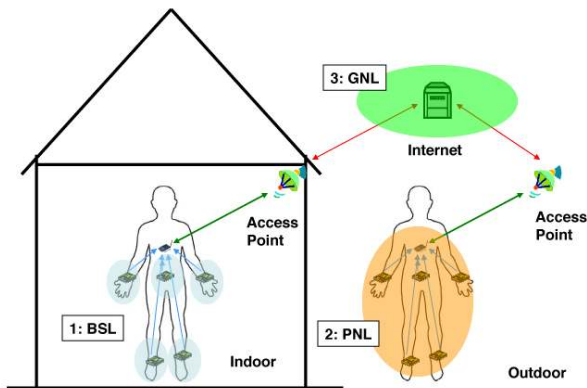
1. *DexterNet* supports an open-source on-node signal processing library, namely, *SPINE* (Signal Processing In Node Environment) [15]. To our best knowledge, *SPINE* is the only open-source library that is versatile enough to support heterogeneous body sensors. As a result, higher-level applications through *DexterNet* can seamlessly control different types of body sensors.
2. Harnessing the rich functionalities in *SPINE*, *DexterNet* supports real-time signal collection and sensor management. The system can be dynamically configured over the air. It provides a set of on-node services that can be tuned and activated by the user depending

\*This work was supported in part by TRUST (The Team for Research in Ubiquitous Secure Technology), which receives support from the National Science Foundation (NSF award number CCF-0424422) and the following organizations: AFOSR (#FA9550-06-1-0244), Cisco, British Telecom, ESCHER, HP, IBM, iCAST, Intel, Microsoft, ORNL, Pirelli, Qualcomm, Sun, Symantec, Telecom Italia and United Technologies. This work was also supported in part by ARO MURI W911NF-06-1-0076, the Center for Information Technology Research in the Interest of Society (CITRIS), and Finnish Funding Agency for Technology and Innovation (Tekes). Corresponding author: P. Kuryloski (pjk25@cornell.edu).

on different application needs.

3. To support long-term monitoring of multiple subjects in both indoor and outdoor environments, DexterNet adopts a flexible three-layer BSN architecture, namely, a body sensor layer (BSL), a personal network layer (PNL), and a global network layer (GNL).

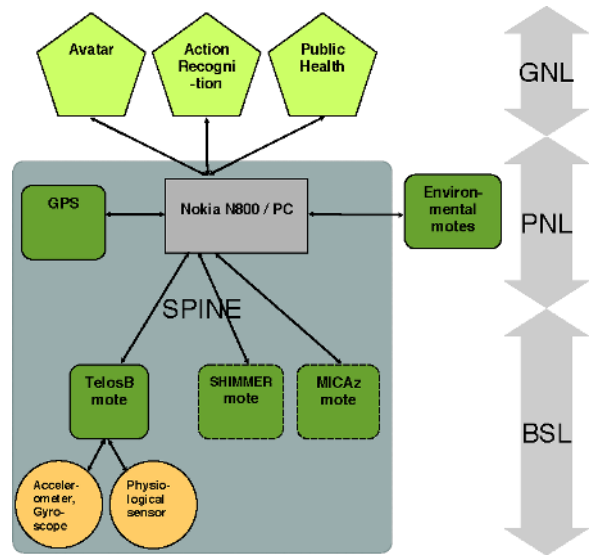
Figure 1 shows the three-layer hierarchy of DexterNet. At the BSL, the system supports a list of commercial and custom-built body sensors. A sensor board then connects with a sensor network mote to form a wearable sensor mote. At the PNL, the body sensors communicate with a wireless mobile station. The SPINE library installed on the body sensors and the mobile station manages the data collection, processing, and transmission of the data, and can be controlled via commands issued from the station.



**Figure 1.** The three-layer hierarchy of the DexterNet system: 1. Body sensor layer (BSL). 2. Personal network layer (PNL). 3. Global network layer (GNL).

DexterNet provides a rich, complete pathway for sensing, distributed processing, wireless communication, and data fusion, which serves as a foundation for higher-level applications at the GNL. We hope that in providing an open platform with DexterNet, variations in functionality can be built using a common base. This will avoid redundant development efforts in different sensor network systems. In addition, the diverse nature of our team has driven the design requirement for DexterNet to provide maximum flexibility and extensibility, with maximum potential for reusability of its components.

**System Architecture.** The architecture of DexterNet is shown in Figure 2. The open-source SPINE framework provides the flexibility in constructing physical components of the system at the BSL and PNL layers. Particularly, SPINE has been developed such that there is a separation in code of its sensing, processing, and communication functions. As a result, SPINE is portable across TinyOS mote platforms, and easily extends to support new sensors through the use of sensor drivers.



**Figure 2.** The architecture of DexterNet. The PNL includes a portable base station (such as the Nokia N800 wireless tablet) and associated sensors. The GNL includes our applications built with the DexterNet system.

Equipped with the versatile three-layer architecture and the open-source on-node library SPINE, DexterNet presents a competitive framework to support a variety of applications in healthcare, military, and consumer electronics. For example, a fall detection function has been implemented at the BSL level using SPINE on-node functions. In our implementation, each motion sensor is capable of outputting a binary decision of a falling event. Such functions reduce the amount of data that has to be transmitted between the nodes and the base station. More sophisticated applications such as human activity recognition can be implemented at the PNL level, which relies on the full-body motion data measured at different key locations of the body. Each of our applications, including avatar visualization, action recognition and asthma studies, is built on top of the SPINE API. They each configure the appropriate sensors and on-node signal processing according to their specific goals and requirements. These applications need not depend on any specific sensor drivers. Furthermore, developers can simultaneously work on application-level software as well as SPINE software without the tight coupling required in traditional application-specific sensor network systems.

## 2 Hardware

The heterogeneity of DexterNet allows a wide variety of sensors and motes to be integrated into the system. DexterNet currently supports three types of sensors that measure physiological parameters, motion of the body, and the location of the wearer using Global Positioning System (GPS). For example, the Intel SHIMMER mote has an onboard accelerometer, MicroSD slot, and ADC converters for attach-

ing external sensors. The MICAz mote has many sensors available as addons, including sensors such as GPS, humidity, barometric pressure, ambient light, sound, magnetometer, etc.

We have recently designed a custom-built motion sensor node with a triaxial accelerometer and a biaxial gyroscope (as shown in Figure 3). The node incorporates a TelosB mote for wireless network communication and on-node signal processing. The mote uses 802.15.4 wireless protocol and can achieve approximately 64 Hz maximal sampling rates when raw measurement data are desired. Simple signal processing tasks can be done in real-time with the on-board MSP430F1611 microcontroller running at 8 MHz clock frequency with 10 kB of internal RAM. The MSP430 also conducts the conversion of analog measurement signals to digital with 12 bit resolution. The capacity of the battery is 600 mAh, which supports maximum continuous measurement and wireless raw data output for approximately 20 hours. One advantage of our motion sensor design is its low cost compared to other commercial systems (e.g., SHIMMER), which often are more expensive and do not necessarily support open-source development.



**Figure 3.** Illustration of the motion sensor node. The sensor board on the top is a custom-built motion sensor with a triaxial accelerometer and a biaxial gyroscope. The middle layer is a Li-ion battery. The sensor board on the bottom is a standard TelosB network node.

In addition, a physiological sensor node [13] has been designed to expand the available functionalities for potential healthcare applications (as shown in Figure 4). There have been multiple BSN projects that measure heart activity (ECG signal) and/or breathing measurement. Unfortunately, the existing systems so far only measure the rate of breathing and neglect a more essential parameter of breathing, namely, the volume. In physiology of human metabolism, the respiration rate is merely an indirect and unreliable indicator of the *minute volume* parameter.

The communication, signal processing, and battery components of the sensor are interchangeable with the respective components of the previous motion mote. The sensor board uses four electrodes connected to the surface of the ribcage. The same electrodes are used for both heart and breathing measurement. Breathing is measured using electrical impedance pneumography (EIP), which involves feeding a very small constant high frequency cur-



**Figure 4.** Illustration of the physiological sensor board (top) connected to the TelosB network node (bottom). The middle layer is a Li-ion battery.

rent through the thorax and measuring the voltage created by the current. Changes in the voltage are proportional to changes in the conductivity of the thorax caused by breathing. Inspiration increases the impedance due to increased low-conductivity air content in the measurement region and anatomical changes in the thorax. Expiration lowers the impedance due to the opposite behavior. The raw measurement signals are interpreted to derive physiological parameters: inspiration-expiration times, tidal volume, minute volume and their associated variabilities.

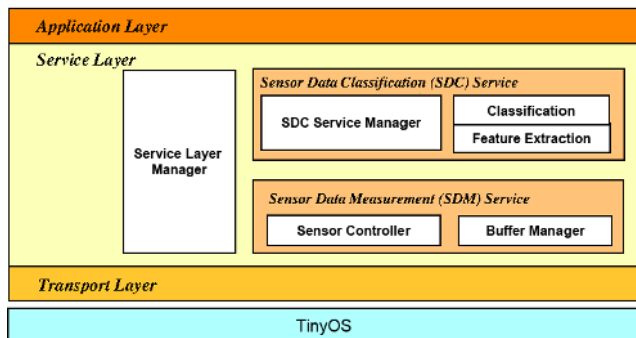
The EIP breathing measurement is challenging to analyze and prone to distortion, partly because the shape of the waveform is important. On the other hand, the ECG signal acquisition is rather trivial, because in heart rate analysis one is mainly interested in detecting the time moments of heart contractions occurring as so-called *R*-peaks. From the raw measurement, the heart rate and heart rate variability can then be derived. With multivariate analysis, the EIP and ECG parameters can be used for a variety of higher level long-term derivations, such as the energy expenditure estimate. Some representative physiological output from the sensor is demonstrated in Figure 6.

### 3 The SPINE Framework

SPINE (Signal Processing In Node Environment)<sup>1</sup> is an open-source framework for distributed signal processing algorithms in wireless sensor networks (WSNs). The functional architecture of SPINE is shown in Figure 5. It provides a set of on-node services that can be tuned and activated by the user over the air depending on different application needs. The open-source framework speeds up the design of WSN applications through high-level abstractions and provides support to quickly explore implementation tradeoffs through fast prototyping. SPINE also provides an efficient wireless communication protocol for dynamic network configuration and management.

The SPINE framework has two main modules, one on the sensor node side, and the other on the base station side. The node module is developed in TinyOS. It provides the

<sup>1</sup>The SPINE software is available for download at <http://spine.tilab.com/>.



**Figure 5.** The SPINE functional architecture.

following three on-node service components: 1. Communication. 2. Sensing. 3. Signal processing.

The communication component serves to arbitrate the radio and to fragment long payloads into multiple outgoing messages as necessary. SPINE has been recently updated to make use of the MAC Layer Architecture (MLA) portion of the Unified Radio Power Management Architecture (UPMA) described in [7]. UPMA provides a set of TinyOS components for developing MAC and power management protocols, and provides the flexibility to choose from several MAC protocols when compiling a TinyOS application. Of those provided by UPMA at this time, which include B-MAC, X-MAC, SCP and 2 TDMA variants, X-MAC and SCP provide the best performance in terms of throughput and duty cycle. While TinyOS 2’s radio low power mode for cc2420 radio equipped motes is an X-MAC variant, UPMA in SPINE allows the flexibility to develop and swap in optimized TDMA or other MAC protocols as necessary based on energy, throughput, or latency concerns.

The sensing module encapsulates active sensors in instances of a common interface, allowing them to be accessed in generic fashion by signal processing components. The interface identifies each sensor, which may have from one to four associated data streams or channels. This, for instance, encapsulates a triaxial accelerometer as a single sensor with three channels. When a new sensor is introduced, a wrapper is built for its driver, at which point it is immediately made available to all existing signal processing features on that node. Since this wrapper does not contain communication or other logic, it can be written rapidly and easily debugged.

The signal processing component provides two different modules, and has been designed to be expandable to include new modules. The first, called the FeatureEngine, periodically performs feature extraction on a sliding window of sensor data and reports it. The length of that window, as well as the number of new samples buffered at a shift, is specified during over-the-air setup. The second module, the AlarmEngine, reports chosen features (specified in same manner as those in the FeatureEngine) conditionally based on some thresholds. Both the FeatureEngine

and AlarmEngine make use of a library of feature extraction functions. Those currently available in the library include Min, Mean, Max, Amplitude, Median, Mode, Range, Standard Deviation, Variance, and Energy.

The server module is implemented in Java SE and acts as the coordinator of a sensor network. It provides an API to control on-node services depending on the application requirements. This has allowed us to deploy the server module on the Nokia N800 tablet, as well as on Mac, Windows, and Linux desktop and laptop PCs. The module can be ported to other portable platforms that support embedded Linux OS and the Java environment (e.g., Motorola E680i). The N800 we choose to demonstrate in this paper provides a platform for Bluetooth and Wi-Fi connectivities to allow forwarding of data to the GNL. It allows the realization of a body sensor network that can operate both inside the home and outdoor, a key feature for supporting a wide variety of human monitoring applications.

## 4 Applications

### 4.1 Body Action Monitoring

Utilizing the motion data from a distribution of body sensors, DexterNet can be engaged in a variety of applications to monitor human actions/activities. We have designed an application called *Avatar*, which uses a network of motion sensors to reconstruct and visualize the wearer’s full-body motion in real time. The application can be used to remotely monitor and assess the well being of elderly people living alone. It can also be used in tele-healthcare for physicians to remotely record and visualize the movements of patients. For the purpose of visualizing motion, a configuration of five nodes (one on each leg, one on each arm, and one on the torso) are the minimum number of sensors required. To provide finer measurement of the full-body movement, more sensor nodes can be worn by a person. Through SPINE, each node estimates the pitch and roll of its orientation in space and reports this pair of values to the base station. The orientation in space of a single sensor node is computed based on the apparent direction of gravity as seen by the sensor board’s accelerometer. When considered as a vector, the accelerometer will read the vector sum of gravity and acceleration resulting in movement of the sensor board. Under relaxed motions, the motion component of the vector is less than 10% of the magnitude of the gravity vector. As a result, this motion component is neglected and we continuously interpret the direction of the accelerometer vector as the direction of gravity.

In addition to using graphical avatars to visualize and analyze human poses and movements, another application is human action/activity recognition [18]. First, we have constructed an open-source benchmark database for human action recognition using DexterNet called *Wearable Action*

*Recognition Database (WARD)*<sup>2</sup>. The database was constructed over the course of two weeks in an indoor environment. The data were sampled from 7 female and 13 male subjects (in total 20 subjects) with ages ranging from 19 to 75. The current version, version 1.0, includes the following 13 action categories: 1. *Stand*. 2. *Sit*. 3. *Lie down*. 4. *Walk forward*. 5. *Walk left-circle*. 6. *Walk right-circle*. 7. *Turn left*. 8. *Turn right*. 9. *Go upstairs*. 10. *Go downstairs*. 11. *Jog*. 12. *Jump*. 13. *Push wheelchair*.

We have studied a distributed recognition algorithm to classify human actions using the low-bandwidth motion sensors, called *distributed sparsity classifier (DSC)* [18]. DSC classifies human actions using a set of training motion sequences as prior examples. It is also capable of rejecting outlying actions that are not in the training categories. The classification is operated in a distributed fashion on individual sensor nodes and a base station computer. More importantly, the algorithm is robust and adaptive to the change of active sensors in a body network on-the-fly due to either sensor failure or network congestion. The recognition precision only decreases gracefully using smaller subsets of active sensors. Table 1 shows the recognition accuracy w.r.t. different configurations of the body sensor network. The numbering of the sensors is: 1. Left wrist. 2. Right wrist. 3. Waist. 4. Left ankle. 5. Right ankle. The two indices are false positive rate (FPR) and verification rate (VR). The high recognition accuracy on the WARD database indicates that DSC should be able to classify other action categories such as bicycling, golfing, and possibly hand motions.

**Table 1.** Accuracy of DSC using up to 5 wearable motion sensors in WARD.

Sen #	1-5	1,3,4	1,4	1,3	3,4
FPR [%]	7.14	8	11.49	17.97	14.63
VR [%]	94.59	96.84	98.19	95.57	97.28

## 4.2 Public Health

DexterNet has many applications within the field of public health, where the ability to objectively monitor the activity patterns of users may improve understanding of exposures to environmental hazards such as air pollution that are associated with asthma attacks, chronic obstructive pulmonary disease (COPD), cardiovascular disease, as well as premature mortality. While others have studied the environment (e.g., air pollution) using participatory sensing approaches, our approach focuses on better understanding when and in what environments personal activities occur and the corresponding physiologic responses to these activities, as measured by a combination of motion, physiological, and GPS sensors. In particular, the addition of

<sup>2</sup>WARD is available for download at: <http://www.eecs.berkeley.edu/~yang/software/WAR/>.

the physiological sensor provides a mechanism to monitor physiological responses to such exposures in real time that may be predictive of severe disease events (e.g., a heart attack or an asthma attack). The inclusion of geographic location data from the GPS is also important to understand the underlying environmental context for such applications.

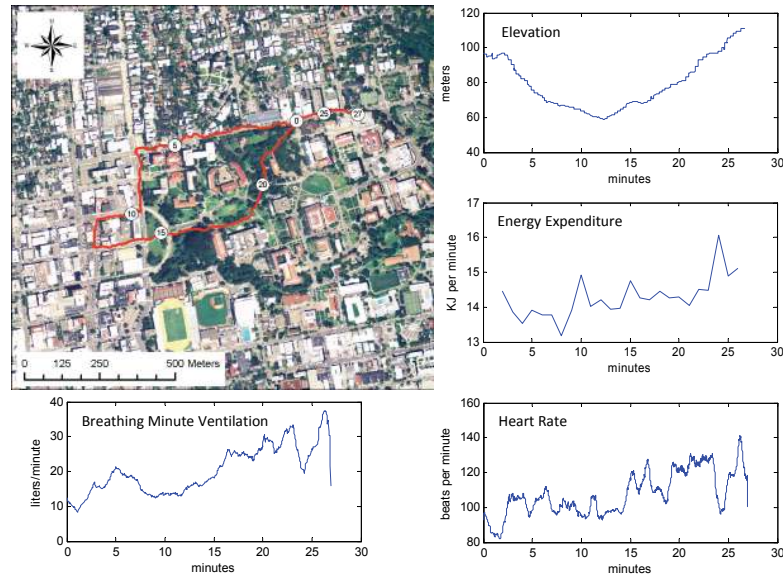
To evaluate the DexterNet system, we have conducted a field experiment in which the system was used to collect and process an integrated set of data related to an individual’s outdoor experience. The experiment consisted of a series of prescribed walks. A convenient sample of six adults (five male and one female) were asked to walk a 2.4 km route. The walk included sections that were uphill, downhill, and flat, as well as sections that were along a busy roadway, a downtown commercial/retail area, as well as a calmer path through a university campus. Over the course of the walk, various sensor data were logged, including triaxial accelerometry and biaxial gyroscopy (at the left wrist, waist, and left ankle positions), and GPS. The motion data were logged at 40 Hz. GPS was logged at 1 Hz. These data were combined and processed to ascertain specific information on the individual’s experience (e.g., assessing the magnitude of physical activity at certain geographic locations). Figure 6 illustrates the GPS trace of the walking route. The application determines the changes in elevation during the walk from the GPS data. The motion sensor at the waist was used to derive the energy expenditure using the Generalized Linear Model [5]. The breathing minute ventilation is derived from the EIP signal [13]. Heart rate is obtained from the ECG signal using a simple *R*-peak detection algorithm.

## 5 Discussion

In this paper, we have discussed DexterNet, a novel platform for heterogeneous body sensor networks. The key attributes of DexterNet are manifold: 1. It promotes an open-source sensor environment that supports on-node computation, robust sensor communication, and online reconfigurable network management. 2. The platform is versatile enough to support a variety of existing body sensors and other future sensors that comply with the SPINE specifications. 3. Through a hierarchy of three network layers, it resolves the dependency of higher-level applications toward the implementation of wireless body sensors and communication protocols.

There are numerous potential services that may be implemented through DexterNet, especially in the area of preventive healthcare. For example, it is possible through the classification algorithms described to identify conditions that are predictive of asthma attacks and warn users to reduce physical activity and/or move indoors. Such systems can also create maps of microscale air pollution when they are deployed in sufficiently large numbers.





**Figure 6.** GPS trace of campus walk with derived information from GPS, motion sensor, and physiological sensor. Circles on the map indicate elapsed time in minutes.

## Acknowledgments

The authors would like to thank Dr. Marco Sgroi at the WSN Lab Berkeley, Dr. Yuan Xue at the Vanderbilt University, and Dr. Roozbeh Jafari at the University of Texas, Dallas, for their valuable suggestions and literature references.

## References

- [1] P. Barralon, N. Vuillerme, and N. Noury. Walk detection with a kinematic sensor: Frequency and wavelet comparison. In *IEEE EMBS Conf.*, pages 1711–1714, 2006.
- [2] J. Burke, D. Estrin, M. Hansen, A. Parker, N. Ramanathan, S. Reddy, and M. Srivastava. Participatory sensing. In *World Sensor Web Workshop*, 2006.
- [3] R. Chakravorty. A programmable service architecture for mobile medical care. In *Pervasive Comp. and Comm. Workshop*, 2006.
- [4] J. Chen, K. Kwong, D. Chang, J. Luk, and R. Bajcsy. Wearable sensors for reliable fall detection. In *IEEE Eng. in Med. and Bio. Conf.*, pages 3551–3554, 2005.
- [5] K. Chen and M. Sun. Improving energy expenditure estimation by using a triaxial accelerometer. *Journal of Applied Physiology*, 83:2112–2122, 1997.
- [6] T. Choudhury, S. Concolvo, B. Harrison, J. Hightower, A. LaMarca, L. LeGrand, A. Rahimi, A. Rea, G. Borriello, B. Hemingway, P. Klasnja, K. Koscher, J. Landay, J. Lester, and D. Wyatt. The mobile sensing platform: An embedded activity recognition system. *Pervasive Computing*, pages 32–41, 2008.
- [7] K. Klues, G. Hackmann, O. Chipara, and C. Lu. A component-based architecture for power-efficient media access control in wireless sensor networks. In *SenSys*, 2007.
- [8] D. Malan, T. Fulford-Jones, M. Welsh, and S. Moulton. CodeBlue: An ad hoc sensor network infrastructure for emergency medical care. In *Workshop on Wearable and Implantable Body Sensor Networks*, 2004.
- [9] A. Milenković, C. Otto, and E. Jovanov. Wireless sensor networks for personal health monitoring: Issues and an implementation. (in press) *Computer Communications*, 2006.
- [10] M. Morón, E. Casilari, R. Luque, and J. Gázquez. A wireless monitoring system for pulse-oximetry sensors. In *Proceedings of the 2005 Systems Communications*, 2005.
- [11] N. Oliver and F. Flores-Mangas. HealthGear: A real-time wearable system for monitoring and analyzing physiological signals. In *Workshop on Wearable and Implantable Body Sensor Networks*, pages 61–64, 2006.
- [12] I. Pappas, T. Keller, S. Mangold, M. Popovic, V. Dietz, and M. Morari. A reliable gyroscope-based gait-phase detection sensor embedded in a shoe insole. *IEEE Sensors Journal*, 4(2):268–274, 2004.
- [13] V.-P. Seppä, J. Väisänen, O. Lahtinen, and J. Hyttinen. Assessment of breathing parameters during running with a wearable bioimpedance device. In *European Congress for Medical and Biomedical Engineering*, 2008.
- [14] A. Sixsmith and N. Johnson. A smart sensor to detect the falls of the elderly. *Pervasive Computing*, pages 42–47, 2004.
- [15] T. S. Team. The spine manual version 1.2. Technical report, Telecom Italia Lab, 2008.
- [16] G. Williams, K. Doughty, K. Cameron, and D. Bradley. A smart fall and activity monitor for telecare applications. In *Int. Conf. in Med. and Bio. Society*, 1998.
- [17] A. Wood, G. Virone, T. Doan, Q. Cao, L. Selavo, Y. Wu, L. Fang, Z. He, S. Lin, and J. Stankovic. ALARM-NET: Wireless sensor networks for assisted-living and residential monitoring. Technical report, Department of Computer Science, University of Virginia, 2006.
- [18] A. Yang, R. Jafari, S. Sastry, and R. Bajcsy. Distributed recognition of human actions using wearable motion sensor networks. (in press) *Journal of Ambient Intelligence and Smart Environments*, 2009.