Context-driven, Prescription-Based Personal Activity Classification: Methodology, Architecture, and End-to-End Implementation

James Y. Xu, *Student Member, IEEE*, Hua-I. Chang, Chieh Chien, William J. Kaiser, *Senior Member, IEEE*, and Gregory J. Pottie, *Fellow, IEEE*

***Abstract*—Enabling large-scale monitoring and classification of a range of motion activities is of primary importance due to the need by healthcare and fitness professionals to monitor exercises for quality and compliance. Past work has not fully addressed the unique challenges that arise from scaling. This paper presents a novel end-to-end system solution to some of these challenges. The system is built on the prescription-based context-driven activity classification methodology. First, we show that by refining the definition of context, and introducing the concept of scenarios, a prescription model can provide personalized activity monitoring. Second, through a flexible architecture constructed from interface models, we demonstrate the concept of a context-driven classifier. Context classification is achieved through a classification committee approach, and activity classification follows by means of context specific activity models. Then, the architecture is implemented in an end-to-end system featuring an Android application running on a mobile device, and a number of classifiers as core classification components. Finally, we use a series of experimental field evaluations to confirm the expected benefits of the proposed system in terms of classification accuracy, rate, and sensor operating life.**

***Index Terms*—Activity monitoring, context driven, wireless health.**

I. Introduction

THE proliferation of powerful mobile devices, along with the rapid advance in microelectronics, has brought microelectromechanical system inertial sensors, low-power processors, ubiquitous computing, and reliable global data networks. This enables advances toward solving urgent problems in health and wellness promotion, diagnostics, and treatment of conditions. In particular, the integration of the state of the art in sensor technology, signal processing, and mobile computing can now enable large-scale monitoring and classification of a range of motion activities, providing evidence-based tools to monitor patient physical exercises for quality and compliance [1], [2]. For example, in chronic diseases such as stroke, physicians have the need to monitor a patient's walking speed in the hospital, and ensure that they are intermittently standing to alleviate deterioration in exercise tolerance [3]. Once they are discharged, at-home physical rehabilitation is central in reducing recurrent stroke and myocardial infarction, dependence on others and the cost of care [1]. Here, physicians prescribe a set of recommended daily exercises to the patients, and need to monitor them for quality and compliance. Inexpensive activity monitoring is a key strategy to providing quantitative metrics.

A large body of work has focused on the accurate detection of physical activities [4]–[9]. However, enabling monitoring in large, diverse user communities has not been addressed. Our recent experience from a large international clinical trial for disabled persons [10] points to unique challenges associated with scaling. First, domain experts such as clinicians and fitness trainers prescribe exercises on a daily basis, but the quality and quantity performed by subjects are not monitored, vastly decreasing the effectiveness. Second, in large-scale deployments, domain experts come from diverse backgrounds with unique sets of activities of interest. As the number of potential motions increase, traditional classifiers suffer from degraded performance and reliability. Third, nonengineering domain experts do not accept complex classification systems requiring their input on training and classifier selection. Furthermore, end users are often impaired, and only the simplest instructions can be used under ample guidance and feedback.

To achieve the goal of enabling large-scale monitoring and classification, we propose a novel end-to-end system that provides context-driven personalized activity classification following a prescription model. Each of the aforementioned unique challenges is addressed in the following novel ways: 1) to allow seamless monitoring of prescribed physical exercises for quality and compliance, we present a prescription service-based methodology; 2) since the diverse user communities require personalized activity monitoring, we propose a context-driven approach where the context is redefined from previous work [11]–[14], and scenarios are defined as a natural extension; 3) a flexible architecture is crafted to provide the roadmap to an end-to-end system, with a management application tailored toward domain experts such as doctors, and a physical package containing sensors bundled with a mobile-based client targeting end users. In this paper, focus is on the following three major challenges: 1) the ability to accurately detect context using multiple sensing modes and machine learning; 2) the use of context

Manuscript received February 13, 2013; revised July 31, 2013; accepted September 13, 2013. Date of publication September 20, 2013; date of current version May 1, 2014. This work was supported in part by the National Science Foundation under Grant. 0120778 (The Center for Embedded Networked Systems) and in part by the Qualcomm Innovation Fellowship #S2012-1638.

The authors are with the Electrical Engineering Department, University of California Los Angeles, Los Angeles, CA 90095 USA (e-mail: jyxu@ucla.edu; hichang@ucla.edu; jaychien@ucla.edu; kaiser@ee.ucla.edu; pottie@ee.ucla.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JBHI.2013.2282812

to restrictively select activities needing classification, reducing the overall classification complexity and improving classification accuracy, speed, and energy usage; and 3) the ability for experts from different domains to individually prescribe sets of physical activities of interest under different environments.

II. Related Work

As this paper relies heavily on machine learning and in particular supervised classifiers, a brief introduction of terminologies and processes involved is provided here with respect to activity monitoring. In general, a classification system includes an algorithm for making decisions (a classifier), input data, output labels from the classifier (classes), data derived from input that can aid the classifier in making decisions (features) and a model that describe the relationship between features and classes (model). The classifier is first presented with a model and a set of input data with annotated classes. This is called a training dataset and is used by the classifier to learn the relationship between features and classes (training the classifier). Once a classifier has been trained, it is then able to utilize the model in conjunction with new input data to produce classification (output a class label) [15]. In activity classification, the input is obtained using sensors such as body worn inertial measurement units (IMUs), the classes can be individual activities such as walking or running; features can be derived from input such as standard deviation of the foot acceleration; and a model can describe how walking and running can be identified using this feature. The system is trained by instructing the person to perform both walking and running activities and providing this data with annotation to the classifier.

Many investigations have demonstrated the benefits of activity monitoring through sensors. One system for measuring home-based physical rehabilitation has been described in [7]. Using a signature detection algorithm and accelerometer's signal vector magnitude as a feature, the system detects if a user has performed a set of rehabilitation exercises accurately, and provides appropriate feedback. In [8], a human activity classification system was developed for promoting exercises in an effort to reduce injuries. The system uses multiple on body accelerometers, and a large number of binary classifiers each trained to recognize specific activities. A series of optimizations link the individual classifiers to produce a final output. In [5], a small set of activities are recognized using a single triaxial accelerometer and machine learning techniques. Hierarchical classification is performed by preliminary clustering of motion into static, transitional, and dynamic states, followed by refined classification of actual activities. Accelerometer sensor data and machine learning algorithms were also applied for monitoring intervention effectiveness of acute stroke patients [4]. The Naïve Bayes classifier is first used to detect activities such as walking, and a dynamic time warping algorithm is then used to compare segments of activity against previous templates. This provides physicians with the ability to directly measure a patient's activity level after discharge, improving upon the surrogate laboratory measurements administered only in a clinical setting. Most methods confront the challenge of classifying a specific motion among many possibilities at any observation time [10]. As the number of potential motions increase, the classifier model complexity increases and classification performance and reliability are degraded. In addition, these systems do not address the issue of rapid adaptation to the demands of large heterogeneous user communities, where different activities are of interest, requiring separate models, classification methods, and features.

The recognition of user and environmental context is a primary capability for improving human–machine interaction and enabling low energy operation while retaining system performance [14]. Studies have emerged in wireless health that attempt to integrate context and activity classification. A middleware for managing context data was presented in [16]. Context was defined in the study as all measurable aspects of a person, including physical activities. The middleware first handles the transmission, reception, storage of context data from sensors, and then, provides a query platform enabling in-community care by healthcare providers. In [17], a multisensor wearable system was proposed that enables motion classification. Thirty sensors were embedded into a garment with multiple processing nodes responsible for distributed processing. This study treated physical activities as contexts, and focused on sensor fusion techniques. Using context data to aid activity classification was probed in [18], where a large number of sensors were placed on subjects to collect context data including ambient light level, electrocardiography and skin temperature. The study extended traditional accelerometer-based classification using the extra data as features directly in the classifier. Considerable work has also been done to facilitate personalization and adaptation [10], [19], with some papers treating activities as another stream of context information, and a few exploring context aided activity classification. The latter experienced limited success due to ambiguity in the definition of context, and a lack of appropriate methodology and system architecture.

In this paper, we separate contexts from physical activities in order to produce a first level hierarchy, and further achieve prescription-based personalized activity classification. Rather than a middleware solution, our approach is an end-to-end system methodology and architecture to address all aspects from prescription to monitoring of activities.

III. Methodology

The methodology proposed is to pursue personalized activity classification under different contexts, using a prescription-based approach. This requires a new definition of context, a novel method to link contexts and activities of interest, and a new model for prescription, personalization, monitoring, and feedback. We deal with these in turn.

A. Context and Scenarios

The context definition by Dey [11] is often used. While powerful, it is not well suited for monitoring physical activities, where in many cases, a context definition contains activities. There are a number of alternative definitions available [12], [13], which, however, still contain physical activities along with other environmental attributes. Therefore, in this study, *a context is defined*

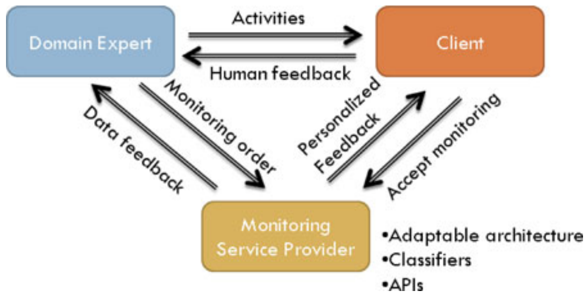


Fig. 1. Prescription model.

TABLE I
EXAMPLE SCENARIOS

Context	Activity Model	Purpose
Patient room	Sitting, Standing, Lying down	Monitor how long a patient has stayed immobile, assess the risk of bed sores and other problems
Rehabilitation	Aerobic exercise, Walking Slow, Walking fast, Fall	Monitor patient's performance in exercises
Hall way	Standing, Walking fast, Walking slow, Fall	Monitor a patient's general physical condition, and detect falls

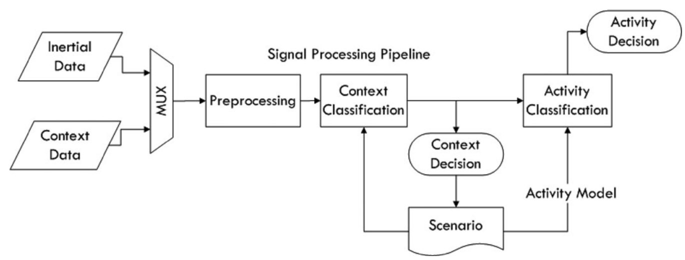


Fig. 2. System high-level description.

as a subset of all attributes that characterizes an environment or situation, external to the user. For example, a "meeting" environment is a context, and its characteristics may involve certain sound profiles and a set of possible locations. "Sitting in a meeting" in contrast is not a context, as it contains the physical activity "sitting."

Personalization is achieved on two levels. First, individuals may have different sets of contexts under which motion classification is required. Then, within each context, there can be a set of individualized activities of interest. This leads to the definition of a scenario: *A scenario is the combination of a context, and a set of activities of interest under the context, with a model for distinguishing the activities.* For example, consider a system that needs to monitor walking, running, and standing when a user is outdoors. The context is outdoors, and under this context, an activity model can use an accelerometer on the ankle with standard deviation in the horizontal direction as a parameter for separating the motions. A number of classification methods such as a Naïve Bayes classifier can then perform activity identification using this model [20].

Compared to dynamic models that can learn over time the activities associated with each context, the scenario approach works well by providing concise definition and tight control for prescription-based monitoring of activities (next section), where experts are interested in prescribing and monitoring activities (such as running) that should be performed under specific contexts (for example, gym).

B. Prescription Model

This paper proposes that a prescription model can be used for context-driven personalized activity monitoring, as shown in Fig. 1. This approach enables healthcare providers to prescribe individualized exercise plans dependent on a subject's needs and monitor them for quality and compliance. A physical package containing sensors and a mobile communication device combined with an application are then sent to the end user. The mobile device with bundled application acts as not only a sensor instrumentation hub, but also a signal processing unit and an avenue for feedback and guidance. Table I lists an example set of scenarios that may be prescribed for the stroke patient example given in Section I.

This system extends well beyond medical usage to any application needing activity monitoring and guidance. For example, fitness trainers can prescribe personalized exercises, their duration and place (gym, home, office) for different clients. The activity monitoring system can then inform the users of their individual training progress.

C. High-Level System Functionality

The following three modes of operation are supported by the system: 1) construction and prescription of models by domain experts; 2) initial training from end users of the classification system to recognize both context and motion; and 3) live monitoring of the end-user's context and motion. High-level functionalities of the system are depicted in Fig. 2. Using various sensors, the system obtains both inertial data that describe motion, and environmental data that describe context. The data flow through a signal processing pipeline and a user's context is first determined. The motion model considered by activity classification after context detection is determined by scenarios. The final output of the system is a user's current context and activity.

IV. ARCHITECTURE

In support of the proposed methodology, a real-time architecture capable of providing subscription service, context-driven activity classification and feedback is designed (see Fig. 3). At the end-user side, a set of sensors is needed with a smart device to provide data to a backend server, where context and activity classification decisions are made. The returned results can be consumed by third party applications. Individual subsystems are modeled as objects, and the entire architecture is defined by a set of interfaces and relationships (see Fig. 4). Each software interface is characterized by its public methods, and defined by its functionality, expected inputs and outputs. By implementing an interface, a class agrees to provide all methods outlined in that interface [21]. Each subsystem can be developed independently without revealing specific implementations, so long as

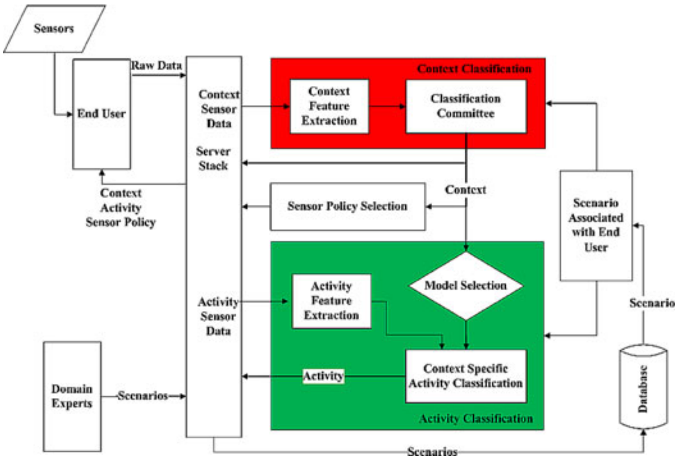


Fig. 3. System architecture.

it implements the required interface. This enables any part of the proposed system to be overridden by custom realizations, allowing for rapid prototyping and evaluation of various algorithms. For example, consider the IContextClassifierEx interface representing a context classifier. Teams can develop optimized, application specific classifiers independently by implementing the interface, and the classifiers can be swapped to adjust the system behavior without affecting other components.

This architecture naturally conforms to the client-server paradigm. Server components include prescription and scenario management, context classification, context-driven activity classification and sensor control. There are two clients: the end-user client allows a user to authenticate, collect required training data for the context and activity classification systems (for the first time use of a scenario), and then, go into live monitoring mode; the domain expert client is used by experts to design and prescribe scenarios for end users.

A. Server Components

1) Context Classification: Our definition of context can capture a large number of situations so that users with different objectives can define their own useful sets. They can identify required characteristics, and select necessary sensors. This generalization increases classification difficulty, as the system must account for a diverse range of data sources such as wireless information, audio, and illumination level. In order to detect context using a variety of data sources, multiple classifiers should be employed for different features. We propose a classification committee consisting of n individual classifiers (see Fig. 5). The individual classifiers are trained separately, and then, tested for individual classification accuracy. A voting weight (α) is determined for each classifier, proportional to the perceived accuracy. When an unknown class is encountered, the committee performs a linear combination of the individual classifiers, and the context with the highest vote is chosen.

Consider a simplified example with a committee consisting of two classifiers for determining if a user is in a quiet office or on the bus. Available data are collected from a smartphone: wireless SSID and signal strength information and audio. The two classifiers in the committee are k-nearest neighbor (kNN) for wireless features and AdaBoost for audio features (Sections II and V, respectively, provide details on the classification process and implementation of these classifiers for the data types in this example). If the unknown data stream contains SSID of the usual workplace with high signal strength and quiet background noise, then both classifiers would output separate decisions that the current context is workplace, and the weighted combination of the committee would be workplace. If the unknown data contained unstable wireless information and high level of background noise containing features common to transports, then the kNN would output "Unknown" with zero weight and the audio classifier would provide the correct context decision.

This committee approach provides flexibility in designing contexts by allowing data fusion of sensors with various data types, and adapts to individuals with varying habits. The classifiers selected for the committee and their implementations can be application specific, each designed to suit the data input available for that application. Outside of the implementations described in this paper, many other combinations can be considered under this architecture.

2) Context-Driven and Personalized Activity Classification: After classifying context, a model used for activity classification is chosen from the corresponding scenario. This is the concept of a context-driven classifier, through which specifically optimized models can be used with each being focused on the activities of interest within a context. Unlike conventional activity monitoring, there is no single list of comprehensive activities that needs to be built into a monolithic classifier. Instead, multiple personalized scenarios are prescribed to a user. There are a number of benefits from this system, such as improved classification accuracy and speed due to model simplification.

3) Sensor Control Policy: By having scenarios describing the contexts and activities of interest, sensor activation schedule and sampling rate can also be optimized to reduce energy demand. For example, there are no upper body motions from the patient room scenario in Table I, and the activities have a low rate of change. It is then safe to disable upper body sensors and reduce sampling rate on lower body sensors with no loss in system performance. The benefits of this are an overall reduction of energy, storage and communication needs. Noticeably, this does not include potential optimization of different radios and transmission modes. The architecture is designed to integrate with consumer products such as mobile phones and off-the-shelf sensors, where Bluetooth is the *de facto* standard form of communication. Future work can integrate radio link energy management with the introduction of Bluetooth Low Energy.

B. End-User Client Components

The end-user client application guides a user in training mode, and displays classification results in online mode. Mobile applications supported by a smartphone are ideal for two primary reasons. First, mobile devices are pervasive, making the client accessible with extensive network infrastructure support. Second, mobile devices are high performance and can act not only as user interface platforms, but also as sensor platforms that

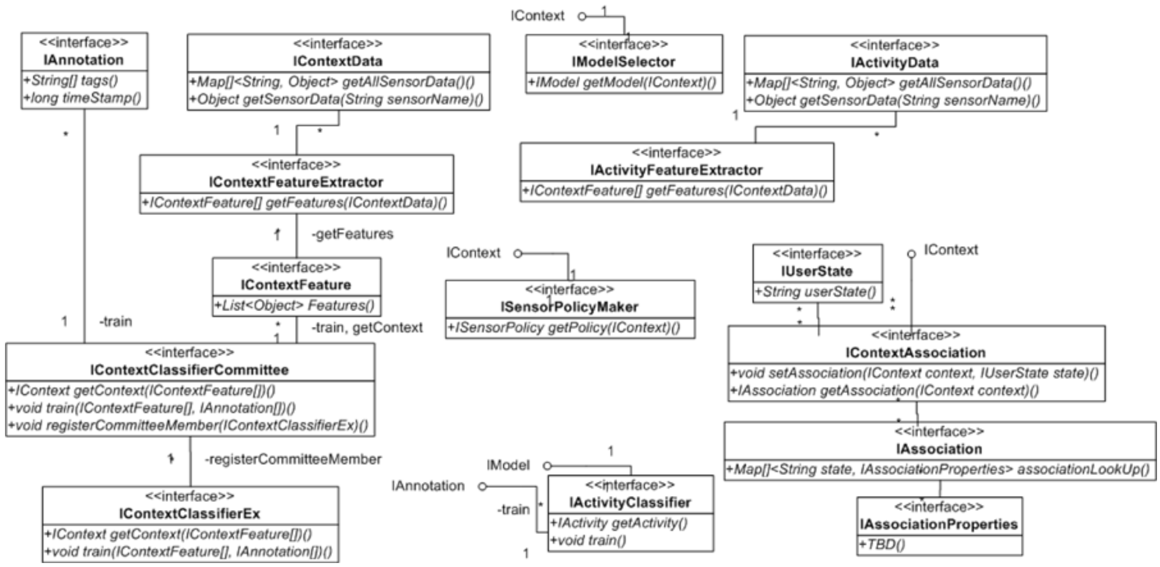


Fig. 4. System interface model.

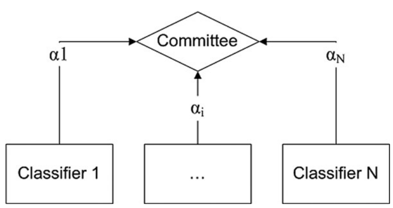


Fig. 5. Classifier committee.

log, process and store data from built-in and external wearable sensors.

1) Sensor Instrumentation: One requirement of the architecture on the end-user client is to reliably obtain data from external sensors. This requirement includes: 1) automatically detect, connect, and control external sensors in real time; 2) continuously track the status of sensors; 3) recover from corrupted data, missing data, and delay. We propose the following air architecture for instrumentation of external sensors (see Fig. 6).

First, the subsystem closest to hardware is the AirInterface. Its implementation should be minimal, supporting only basic read and write operations required for sensor control and data recording. This ensures that the subsystem executes at maximum speed. Attached to the air interfaces are monitors. As the sensors being instrumented can be different, one monitor per interface is necessary. The monitor tracks a sensor's state, notifies the upper layer of changes, and takes appropriate actions autonomously. For example, if a sensor disconnection event is detected, the monitor could notify the upper layer about the disconnection, while trying to reestablish connection through the air interface it is attached to.

Each AirInterface obtains data from a stream established to the target device, and stores them in a shared buffer. A data processor unit runs in parallel to all air interfaces and processes the buffer, synchronizing data from multiple sensors. This is a standard producer–consumer pattern, where the processing unit is decoupled from the recording units through a buffer. This

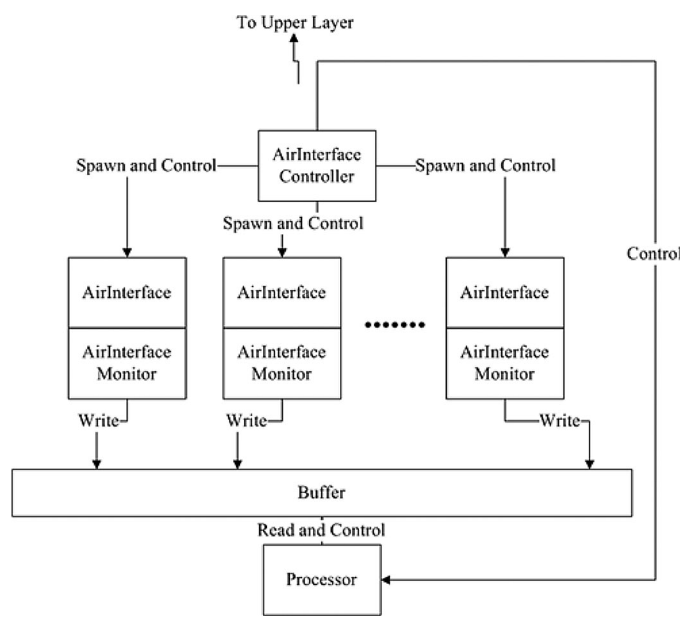


Fig. 6. AirInterface architecture.

buffer also grants protection against spurious delays of sensors, with the tradeoff of some initial delay when the sensors are first turned ON.

Abstracting the underlying sensor instrumentation is the AirInterface Controller. It offers upper layers the ability to initiate connection to sensors, and to obtain synchronized data. Fig. 7 shows the interface model of the overall sensor instrumentation component. Only the controller needs an interface for abstracting with upper layers, with messages marked by DataArrived and Notification interfaces, indicating the availability of synchronized data and special sensor events, respectively.

2) System Training: For the system to work across a large population, individualized training of classifiers is required. When an end user first receives the sensors, they login through

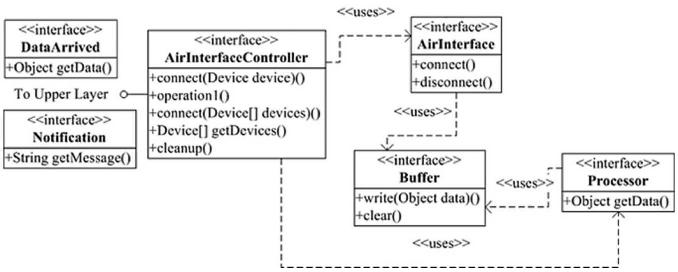


Fig. 7. AirInterface controller interface model.

an application and obtain the scenarios prescribed to them. The scenarios determine what activity and context data need to be collected from the user for classifier training. The application guides the user through training both context and activity classifiers by providing visual cues instructing the user to perform the prescribed activities under corresponding contexts. While these activities are being performed, data are collected from the inertial sensors (activity data) and from the mobile device (context data). Once each scenario has been performed, the context classifier committee is trained to be able to detect all of the prescribed contexts, and the activity classifier is trained for each scenario by training their respective activity model. The length of time required for training each scenario is 5 min in the particular context and 2–3 min for each activity. After training, the end-user client can go into live mode, where data are collected and sent autonomously to a server, and a continuous live stream of context and motion classifications is returned.

C. Domain Expert Client

A domain expert must prescribe to the end user a set of scenarios that specifies the context specific activities to be monitored. This is done through the domain expert client. As experts are likely to be nonengineering professionals, the main focus of this client is to abstract much of the classification system such as models, features, and classifiers. Ideally, the application should offer intuitive drag-and-drop like functions with labels that have explicit meaning such as walking, running.

V. System Implementation

A. Server Implementation

The server implements the core classification components (context and activity), and also handles user and scenario management.

1) User and Scenario Management: User and scenario management on the server leverage standard web technologies that enable secure authentication and transfer of scenario files from domain expert clients to the server, and then, to the end users. Using a representational state transfer (RESTful) web service architecture, we take advantage of existing hypertext transfer protocol (HTTP) and industrial standard secure HTTPS infrastructure [22]. This approach requires a number of key components: a web server for providing the HTTP infrastructure; a platform for developing the web service that enables file transfer; and a naming authority for redirecting requests to the web service. For the server, an Apache web server stack is deployed, which enables web services developed in PHP, and the naming authority in htaccess. The implementation uses a flat-file database system, where data are stored in regular files on a hard drive. The domain experts have privileges to view a list of users belonging to them, and prescribe scenarios. When a new scenario is posted, the server receives the untrained scenario file and saves it in the target user's directory. End users only have privileges to view and use a list of scenarios linked to them.

2) Context Detection: Context detection is performed using separate classifiers on different features using the committee approach from Section IV. In this implementation, the committee is made up of three classifiers: kNN with time as a feature; kNN with wireless media access control (MAC) address and signal strength as features; and AdaBoost with audio peak frequency, peak energy, average power, and total energy as features. These features are extracted from raw sensor data through a java program implementing the *IContextFeatureExtractor* interface (see Fig. 4). As mentioned previously, the choice and implementation of the classifiers are application specific. While this implementation is effective for this study, future work could evaluate the suitability of other techniques such as decision trees and support vector machine (SVM).

The kNN classifier is an instance-based lazy learner that simply stores the training data given [15]. When an unknown class is encountered, the classifier looks for the k-nearest training samples to the unknown class, and a decision is made based on a majority vote. Other than implementation simplicity, another major advantage of kNN is the ability to handle nominal data through custom designed distance functions. This is particularly important for data types such as wireless MAC address values. For our implementation, the kNN with time feature uses a simple absolute distance function that computes the number of seconds between two times, and the kNN with wireless features uses a custom distance function that looks for the closest k labels with overlapping MAC address sets, ranked by signal strength [23].

AdaBoost is a metalearner to be used in conjunction with multiple base learners (weak classifiers). It can combine an ensemble of weak classifiers into a strong one [24]. For our implementation, the AdaBoost.MH algorithm with a decision stump base learner was used [24].

3) User Activity Classification: Two methods for user activity classification were implemented, one being a Bayesian networks (BNs) approach [15], [23] and the other the Wireless Health Institute Sensor Fusion Toolkit (WHISFT) [4], [25].

The BN classifier is built on top of a probabilistic graphical model made up of nodes and edges. The nodes represent random variables (features and the output classes) in the underlying model, and the edges represent conditional dependence between variables. Classification can be performed by populating the graph with observed variables (features) and computing the conditional probability of output classes given the observed variables. The BN approach has the advantage of being user friendly, as domain experts can draw class nodes for the activities of interest, and features that best describe them. This visualized model can serve directly as the BN model for classification of activities by querying class nodes. We implemented

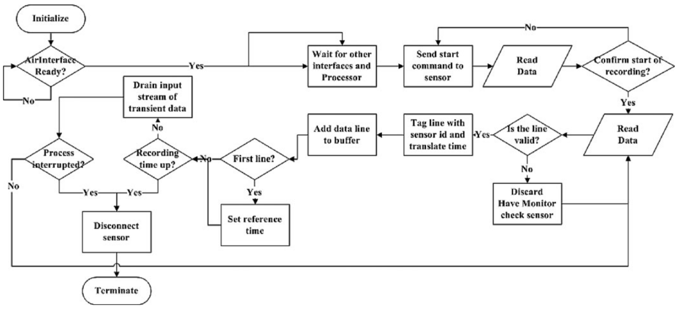


Fig. 8. AirInterface implementation flowchart.

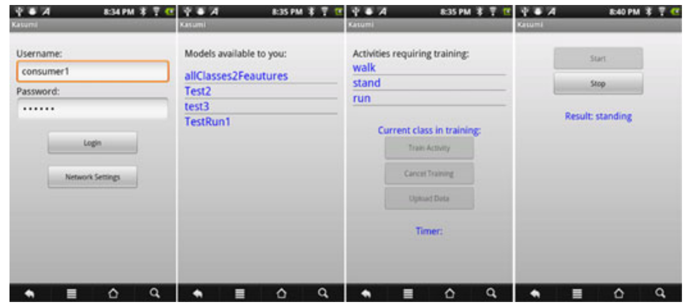


Fig. 9. End-user client.

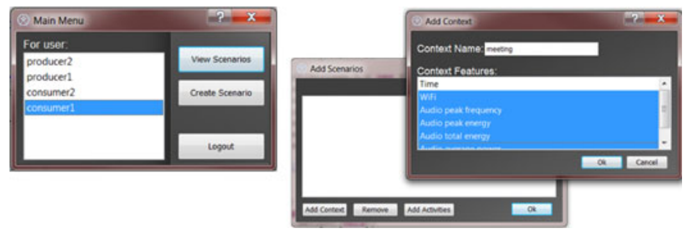


Fig. 10. Domain expert client.

an augmented BN with Dirichlet densities for parameter learning and incorporating expert's expertise, and Pearl's message passing algorithm for inference [15], [23]. However, due to the discrete nature of common BN approaches, the continuous features derived from sensor data had to be discretized. This process significantly reduced classification accuracy, and so the BN approach was not chosen for the final system.

The WHISFT on the other hand is a suite of accurate classification methods for user activities that has undergone testing in diverse situations and clinical settings [4], [5]. It provides multimodal hierarchical classification based on a set of classifiers such as Naive Bayes and SVM [25]. Starting with raw data from multiple sensors, the WHISFT combines streams of data into a single structure. Features such as short time energy, mean, and variance are computed from the combined data structure. There are a number of diverse features, providing freedom in selecting the ones that best suit each application. From the selected features, hierarchical structures can be built to model the classification problem. These are trees that first grossly separate activities into groups sharing similar features, and then, further isolate activities within each group using additional features until all can be identified. At each level of the tree, the WHISFT uses either a Naïve Bayes or SVM classifier to separate unknown data into one of the branches. The final classification result is produced when a leaf node is reached.

B. End-User Client Implementation

1) Sensor Instrumentation: Both inertial data and context data need to be collected by the end-user client. Context data includes sound, time, and wireless information, and are provided by an Android device. In this study, inertial data are provided by nine degrees of freedom Sparkfun Razor IMU that are instrumented by AirInterface components (see Section IV). Each Razor IMU contains a triaxial accelerometer, gyroscope, and magnetometer. Our custom firmware reports relative timestamp since the sensor start time, and synchronization between multiple sensors is carried out by the AirInterface, where a snapshot of relative timestamps of every sensor is taken at a specific time (t), and subsequent data are tracked with t being the zero reference. Sensor timing drift is avoided in the controller implementation by resynchronizing every n seconds. The flowchart (see Fig. 8) depicts a realization of the AirInterface architecture for data collection.

As individual sensors can send data at up to 250 Hz (f), the data processor implements a lookup table data structure with insertion time $O(1)$, at the expense of memory footprint $O(nft)$, where n is the number of sensors, and t is the number of seconds between resync. At every resynchronization, the processor compiles the table into synchronized data frames and makes it available for upper layers through the DataArrival interface (see Fig. 7).

2) Data Acquisition, Training, and User Interfaces: After an end user authenticates with the server [see Fig. 9(a)], the client application displays a list of available scenarios for selection [see Fig. 9(b)]. If the selected scenario is not trained, then the application determines activities present in the underlying model (for example, running, walking), and guides the user through training [see Fig. 9(c)]. For each activity, a 3 min session is recorded, and the data are returned to the server. Once the scenarios have been trained, the client would automatically enter live mode [see Fig. 9(d)], where sensors are instrumented, and data sent back to the server every 4 s for classification.

C. Domain Expert Client Implementation

The domain expert client application allows domain experts to design and prescribe scenarios. Recall that a scenario is made up of a number of contexts and activities, each with its own model. To create a scenario, an expert first generates context models by selecting from a list of prebuilt contexts. Within a context, the expert then defines a list of activities of interest following a similar approach for creating context. Finally, the created scenario and prescriptions are submitted to the server via its RESTful web services. Fig. 10 briefly demonstrates the application using a series of screenshots. The application was built with ease of use in mind and was used successfully by collaborating clinicians.

TABLE II
SCENARIOS

	Walk Around	Walk Normal	Walk Upstairs	Walk Downstairs	Sitting Straight	Sitting Slouch	Stand	Write	Type	Eat	Sleep	Run	Cycle
Home	X				X	X				X	X		
Lab	X				X			X	X				
Cafeteria	X				X		X			X			
Outdoors		X	X	X	X		X					X	
Class	X				X			X					
Bus					X		X						
Gym		X			X							X	X
Library	X				X		X	X					

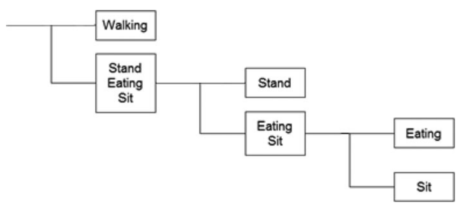


Fig. 11. WHISFT model for cafeteria.

TABLE III
CONTEXT CLASSIFIER ACCURACIES

	AdaBoost	Time kNN	Wireless kNN	Committee
Home	100	91	100	100
Lab	78	68	98	95
Cafeteria	100	0	80	100
Outdoors	81	57	56	72
Class	81	43	95	91
Bus	100	23	30	95
Gym	64	9	93	84
Library	59	0	100	94

VI. System Evaluation

A. System Deployment

Three major components of the system need to be deployed for data collection and system evaluation. The server responsible for user and scenario management, context, and activity classification was deployed to the University of California Los Angeles (UCLA) medical network servers. The domain expert client was given to the collaborators at the UCLA's Department of Neurology. The end-user component is a physical package containing four IMUs with Velcro attachments, a Nexus seven tablet and the mobile application as described in Section V-C. These packages were either given to the subjects in person or mailed to them.

B. Data Acquisition

Table II lists all models used for the experimental trial. In the table, the activity "walking around" refers to nonsustained walking segments that are typical of walking in confined spaces, while "walking normal" refers to sustained long distance walk typical of open spaces. Fig. 11 shows an example of the context specific activity model used in WHISFT for the "Cafeteria" context. The activities are on leaf nodes, laid out in a hierarchy. At each branch, a Naïve Bayes classifier makes the branching decision using features in the model.

For data acquisition, 14 subjects each carried a Nexus seven tablet running the Android client and four 9-DOF devices were placed on dominant wrists, knee, ankle, and mid waist. Each subject spent 30 min in each context, and performed every required activity under that context for 2–5 min. The data were then split into training (30%) and testing (70%) sets and tenfold cross-validation was performed to obtain the classification results.

C. Results

1) Context Classification: Table III summarizes the accuracies of the classifier committee in percentage of correctly classified instances. The table also breaks down the committee results into the constituent individual classifiers.

While none of the individual classifiers performed well for all of the contexts, the combined committee of the three classifiers following the approach outlined in Section IV was able to achieve higher accuracy for all contexts. Wireless kNN performs with insufficient accuracy for bus and outdoors. In the bus context, the sensor system detected a large number of wireless access points that had not been incorporated into prior training due to the route of the bus. In the outdoor context case, the system tended to detect access points that belonged to one of the contexts at nearby indoor locations. For example, walking near a building caused the context to be classified as that of a context inside the building. Time kNN is also not sufficiently accurate for a number of contexts, and this is due to the varied nature of when subjects visit these contexts. AdaBoost using sound features seemed to perform well for most contexts, but there were cases where misclassification occurred due to a bus driving nearby or due to long periods of silence that are present in all contexts. There is also negligible overhead observed for the committee approach versus using individual classifiers, as the committee simply performs a linear combination of the individual results.

2) Context-Driven Personalized Activity Classification: Table IV gives the results of context-driven activity classification, where the "Generic" column has results from a standard classification tree (hierarchical Naïve Bayes) using WHISFT, and the "Specific" column has context-driven classifier results. In

TABLE IV
CONTEXT-DRIVEN ACTIVITY CLASSIFICATION ACCURACIES

Context		Generic	Specific	Improve	Context		Generic	Specific	Improve
Home					Class				
	Sleeping	0	94.45	94.45		Walking Around	100	100	0
	Slouching	82.8	99.03	16.23		Writing	34.22	90.27	56.05
	Eating	92.11	92.78	0.67		Sitting	100	98.76	-1.24
	Walking Around	94.34	100	5.66					
	Sitting	80.32	89.78	9.46					
Lab					Bus				
	Sitting	66.9	88.56	21.66		Sitting	50.78	94.89	44.11
	Walking Around	93.82	100	6.18		Standing	75.97	95.76	19.79
	Typing	0	93.43	93.43					
	Writing	32.89	37.22	4.33					
Cafeteria					Gym				
	Standing	100	100	0		Cycling	90.7	98.43	7.73
	Walking Around	93.89	99.02	5.13		Running	100	100	0
	Sitting	60.89	92.23	30.34		Walking	100	100	0
	Eating	88.97	94.89	5.92		Sitting	83.78	92.78	9
Outdoors					Library				
	Walking	93.87	91.83	-2.04		Sitting	76.21	97.67	21.46
	Running	85.82	100	14.18		Walking Around	94.2	95.02	0.82
	Upstairs	60.34	95	34.65		Standing	100	100	0
	Downstairs	60.94	70.89	9.95		Writing	40.9	77.82	36.92
	Standing	100	100	0					
	Sitting	74.43	97.12	22.69					

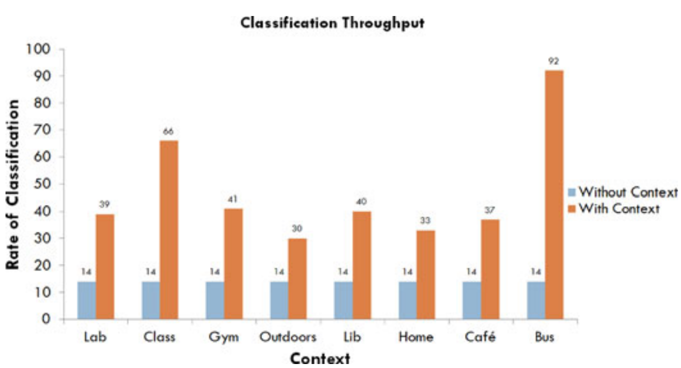


Fig. 12. Classification throughput.

TABLE V
SENSOR REQUIREMENTS

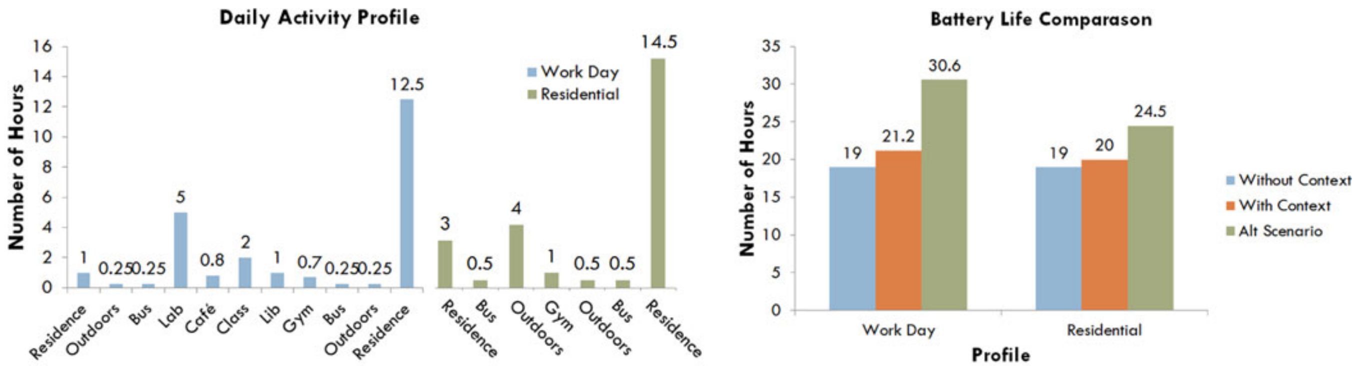
Context	Right Ankle	Knee	Waist	Wrist
Home		X	X	X
Lab	X			X
Cafeteria	X		X	X
Outdoors	X	X	X	X
Class	X			X
Bus			X	
Gym	X	X		
Library	X		X	X

half of the activities monitored, there is a substantial increase in classification accuracy resulting from the context-driven classification, as targeted models with fewer activities and features are presented to the classifier. In the case of typing, writing and eating, a large increase in accuracy can be seen. There are two instances where the context specific model accuracy decreased. However, the decrease is small and can be reduced or eliminated with more training data or model adjustment (both of which are significantly easier to achieve in a smaller, context specific model).

3) Activity Classification Speed Increase: The context-driven classification also offer advances in computational throughput that can enable concurrent real-time classification for a large population. Fig. 12 shows the advance in computational speed that was achieved, where the rate in number of classifications per second is plotted. In all cases, there is a significant decrease in classification time, which indicates that context-driven classification can enable a large online system capable of computing multiple subject's motion.

4) System Energy Usage Improvements: Context-driven classification also offers the capability for selecting optimal sensors and schedules for energy and operating lifetime improvements. This permits a minimum number of sensors to be active while maintaining classification accuracy. Based on scenarios tested (see Table II), a sensor requirement chart was produced (see Table V), where blank cells indicate that a sensor can be safely turned OFF without affecting the accuracy for a scenario.

For example, one scenario prescribed to the user contains the context bus, and under that context, an activity model includes only standing and sitting. This can be easily determined by a waist worn sensor, thus other sensor can be safely switched OFF. Using this table, the sensor policy selector (*ISensorPolicyMaker*) can determine which sensors need to be shut down. To estimate energy reduction, analyses are directed at determining the improvement in operating time by adopting sensor activation schedules, as determined by context. To indicate the improvement over a range of subject behaviors, two cases were taken as examples: residential and work. The typical profiles of their daily life are shown in Fig. 13(a) with the x-axis starting at 8 A.M.. The total operating time using continuous sensor activation, in comparison to context-driven sensor activation, is shown in Fig. 13(b). Results indicate the potential benefits of context-driven sensor energy management. This benefit depends largely on the activities being monitored under the longer contexts.



(a) (b)

Fig. 13. User profiles and their battery life comparison.

VII. Conclusion

Activity monitoring is a valuable tool for disease intervention and guidance in healthcare and wellness promotion. Enabling large-scale monitoring and classification of a range of motion activities is of primary importance due to the need by healthcare and fitness professionals to monitor exercises for quality and compliance. To address the challenges with scaling to large communities, we described the design, implementation, and evaluation of a novel end-to-end system that integrates context into activity classification, with a prescription service.

On the methodology level, we first presented a definition of context and scenario. A prescription model that made use of scenarios then followed that provided personalized activity classification. On an architectural level, we interfaced with wireless sensors, employed a classification committee approach for detecting context of diverse forms, and demonstrated how any current classification system can take advantage of the new context-driven approach through the concept of a context-driven classifier. The architecture also made use of an interface model for software deployment, consequently, providing great system flexibility. We realized the architecture in software, where an Android client application was used to solve issues relating to robust data acquisition and large campaign support. AdaBoost, kNN. WHISFT and BN classifiers were used for both context detection and activity classification, demonstrating the inherent system flexibility. Finally, we evaluated the system using a series of field trials and confirmed its advantages in terms of classification accuracy, computational throughput, and functionality in controlling the activation and selection of sensors.

Many additional research challenges remain to be solved and some areas of this study can be improved. First, the use of dynamic models that can learn the activities associated with contexts in conjunction with scenarios would produce a much more flexible and powerful system that allows both the prescription of targeted activities and the monitoring of generic activities not bound to contexts. In the case of a large population deployment, efforts required to perform system training needs to be reduced by introducing default models that include population norms (thus, less individual training), and additional strong context data such as GPS can be introduced. Finally, while the evaluation provided in this paper demonstrates the effectiveness, efficiency, and potential of the methodology and end-to-end system, a larger study needs to be performed to ascertain the robustness of the system and improve privacy, security, and user friendliness.

Acknowledgment

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation or Qualcomm.

References

[1] B. H. Dobkin and A. Dorsch, "The promise of mHealth: Daily activity monitoring and outcome assessment by wearable sensors," *Neurorehab. Neural Repair*, vol. 25, pp. 788–798, 2011.

[2] J. E. Lim, O. H. Choi, H. S. Na, and D. K. Baik, "A context-aware fitness guide system for exercise optimization in U-health," *IEEE Trans. Info. Tech. Biomed.*, vol. 13, no. 3, pp. 370–379, May 2009.

[3] M. F. Gordon, "Physical activity and exercise recommendations for stroke survivors," *Stroke*, vol. 35, pp. 1230–1240, 2004.

[4] B. H. Dobkin, X. Xu, M. Batalin, S. Thomas, and W. J. Kaiser, "Reliability and validity of bilateral ankle accelerometer algorithms for activity recognition and walking speed after stroke," *Stroke*, vol. 42, pp. 2246–2250, 2011.

[5] B. Logan and J. Healey, "Sensors to detect the activities of daily living," in *Proc. IEEE 28th Int. Conf. Eng. Med. Biol.*, 2006, pp. 5362–5365.

[6] M. Philipose, K. P. Fishkin, M. Perkowitz, D. J. Patterson, D. Fox, H. Kautz, and D. Hahnel, "Inferring activities from interactions with objects," *Pervasive Comput.*, vol. 3, pp. 50–57, 2004.

[7] C. Chiang, Y. Lee, C. Hsieh, C. Chan, and S. J. Hsu, "Quantification of home rehabilitation exercise for the elder's physical fitness monitoring," in *Proc. 5th Int. Conf. Bioinform. Biomed. Eng.*, Wuhan, China, 2011, pp. 1–4.

[8] O. Banos, M. Damas, H. Pomares, and I. Rojas, "Recognition of human physical activity based on a novel hierarchical weighted classification scheme," in *Proc. Int. Joint Conf. Neural Netw.*, San Jose, CA, USA, 2011, pp. 2205–2209.

[9] A. M. Khan, Y. Lee, S. Lee, and T. Kim, "A triaxial accelerometer-based physical activity recognition via augmented signal features and a hierarchical recognizer," *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 5, pp. 1166–1172, Sep. 2010.

[10] X. Xu, M. Batalin, B. H. Dobkin, and W. J. Kaiser, "Robust hierachical system for classification of complex human mobility characteristics in the presence of neurological disorders," in *Proc. Int. Conf. Body Sensor Netw.*, Dallas, TX, USA, 2011, pp. 65–70.

[11] A. K. Dey, "Understanding and using context," *Pers. Ubiquit. Comput.*, vol. 5, pp. 4–7, 2001.

[12] T. Hofer, W. Schwinger, M. Pichler, G. Leonhartsberger, and J. Altmann, “Context-awareness on mobile devices: The hydrogen approach,” in *Proc. 36th Ann. Hawaii Int. Conf. Syst. Sci.*, 2002, pp. 292–302.
[13] P. Prekop and M. Burnett, “Activities, context and ubiquitous computing,” *Special Issue Ubiquit. Comput. Comput. Commun.*, vol. 26, pp. 1168–1176, 2003.
[14] A. Schmidt, M. Beigl, and H. Hans-W, “There is more to context than location,” *Comput. Graph.*, vol. 23, pp. 893–901, 1999.
[15] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques (Morgan Kaufmann Series in Data Management Systems)*, 2nd ed. San Mateo, CA, USA: Morgan Kaufmann, 2005.
[16] H. Pung, T. Gu, W. Xue, P. Palmes, J. Zhu, W. Ng, and C. Tang, “Context-aware middleware for pervasive elderly homecare,” *IEEE J. Sel. Areas Commun.*, vol. 27, no. 4, pp. 510–524, May 2009.
[17] K. Van Laerhoven, A. Schmidt, and H. W. Gellersen, “Multi-sensor context aware clothing,” in *Proc. 6th Int. Symp. Wearable Comput.*, 2002, pp. 49–56.
[18] J. Parkka, M. Ermes, P. Korpipaa, J. Mantyjarvi, J. Peltola, and I. Korhonen, “Activity classification using realistic data from wearable sensors,” *IEEE Trans. Inf. Technol. Biomed.*, vol. 10, no. 1, pp. 119–128, Jan. 2006.
[19] L. Han, S. Jyri, J. Ma, and K. Yu, “Research on context-aware mobile computing,” in *Proc. 22nd Int. Conf. Adv. Inf. Netw. Appl. Workshops*, 2008, pp. 24–30.
[20] Monash University. (2011, Aug. 17). CSE5230 Tutorial: The Naive Bayes Classifier. [Online]. Available: http://www.csse.monash.edu.au/courseware/cse5230/2004/assets/naivebayesTute.pdf
[21] E. Gamma, R. Helm, R. Johnson, and J. M. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Reading, MA, USA: Addison-Wesley Professional, 1994.
[22] J. Y. Xu, “Web-based billing system exploits mature and emerging technology,” *IEEE IT Prof.*, vol. 13, no. 2, pp. 49–55, Mar./Apr. 2011.
[23] J. Y. Xu, “Context guided and personalized activity classification system for wireless health,” M.S. Thesis, Dept. Electr. Eng., Univ. California, Los Angeles, CA, USA, 2011.
[24] R. E. Schapire and Y. Singer, “Improved boosting algorithms using confidence-rated predictions,” *Mach. Learn.*, vol. 37, pp. 297–336, 1999.
[25] C. Chien and G. Pottie, “A universal hybrid decision tree classifier design for human activity classification,” in *Proc. Int. Conf. Eng. Med. Biol. Soc.*, Osaka, Japan, 2012, pp. 1065–1068.

Authors’ photographs and biographies not available at the time of publication.