Portable Wireless Sensors for Object Usage Sensing in the Home: Challenges and Practicalities

Emmanuel Munguia Tapia, Stephen S. Intille, Kent Larson

House_n, Massachusetts Institute of Technology
1 Cambridge Center, 4FL
Cambridge, MA, 02142 USA
{emunguia, intille}@mit.

Abstract. A low-cost kit of stick-on wireless sensors that transmit data indicating whenever various objects are being touched or used might aid ubiquitous computing research efforts on rapid prototyping, context-aware computing, and ultra-dense object sensing, among others. Ideally, the sensors would be small, easy-to-install, and affordable. The sensors would reliably recognize when specific objects are manipulated, despite vibrations produced by the usage of nearby objects and environmental noise. Finally, the sensors would operate continuously for several months, or longer. In this paper, we discuss the challenges and practical aspects associated with creating such "object usage" sensors. We describe the existing technologies used to recognize object usage and then present the design and evaluation of a new stick-on, wireless object usage sensor. The device uses (1) a simple classification rule tuned to differentiate real object usage from adjacent vibrations and noise in real-time based on data collected from a real home, and (2) two complimentary sensors to obtain good battery performance. Results of testing 168 of the sensors in an instrumented home for one month of normal usage are reported as well as results from a 4-hour session of a person busily cooking and cleaning in the home, where every object usage interaction was annotated and analyzed.

1 Introduction

Many ubiquitous computing researchers desire a low-cost kit of stick-on wireless sensors that transmit data indicating whenever various objects in the environment are being touched or used. Availability of such a kit might aid ubiquitous computing and ambient intelligence research efforts on rapid prototyping, context-aware computing, and ultra-dense object sensing, among others. For example, prior work has demonstrated that sensors such as magnetic reed switches [1-3] and RFID tags [4] unobtrusively attached to many objects in an environment can enable a computer system to infer contextual information about the home occupant's movement and everyday activities such as "cooking," "making tea," "vacuuming," and others. Furthermore, the same types of sensors may be useful for studying behavior in non-laboratory environments, providing designers and ethnographers with new data gathering tools [5]. Sensors that can recognize object usage in everyday environments would also foster the development of embedded health assessment applications where

the well-being of individuals living alone in their own homes might be assessed in real-time, providing peace of mind to relatives and loved ones.

But first, in order to collect the necessary data for developing these applications, researchers require an object usage sensing technology that allows them to easily enter an existing home, unobtrusively attach sensors to as many objects as they wish (possibly hundreds), and to leave those sensors deployed for studies that might last months. For this to happen, the object usage sensing technology needs to be wireless, small, easy to install and remove, affordable, robust (electronically and physically) and not necessarily dependent upon a person wearing a tag or reader. People may be unwilling or unable to wear any technology in their homes (e.g. those suffering from medical conditions). These same sensor characteristics will also be desirable once the sensor technology migrates from use in research to use in commercial ubiquitous computing applications.

At first glance, the problem of recognizing object usage from object motion might appear trivial. One might think that a wireless object usage sensor could be designed by simply attaching a low-power, low sampling rate (e.g. 0.5Hz) motion, vibration, tilt, or accelerometer sensor to a wireless node. The device would transmit an object ID wirelessly when significant movement is observed by simply thresholding the motion, vibration, tilt, or acceleration signal, where the threshold is determined in an object independent manner (since setting object-dependent thresholds in the field increases installation complexity). We thought this would be the case.

However, when we deployed such a system in a real home, we observed some challenges that we did not initially fully appreciate. For example, the manipulation of an object can induce vibrations and movement in nearby objects that can cause false positive activations. This happens, for example, when someone forcefully closes a drawer or drops heavy objects on a surface supporting other objects that also have sensors. We refer to this problem as "the adjacency problem". While we realized adjacency activations would occur, the extent to which the problem was experienced in a real home was surprising. These false positive activations not only reduce the battery life of the sensor nodes (due to unnecessary wireless broadcasts) but also may negatively affect the performance of ubiquitous computing applications that depend upon the sensor data.

The second anticipated but underestimated challenge is that the magnitude and temporal characteristics of the motion, acceleration, vibration, or tilt/rotation generated by household objects is dramatically different among objects. Many of these motions are difficult to detect from vibration and tilt switches with low sensitivity and orientation dependencies (blind spots), and even from accelerometers sampling at less than 10Hz. For example, the motion patterns generated by an electric can-opener are different from those generated by closing a drawer on a smooth sliding mechanism, and even though a low sampling rate accelerometer will detect the can opener reliably, we found that sampling rates of at least 20Hz are required to reliably detect the "instantaneous" start and stop of smooth drawers, which dampen the motion to a surprising degree. The typical amount of motion observed when a person sits on a couch is another example of a subtle motion signal that requires a sensitive threshold and high sampling rate. Unfortunately, these relatively high sampling rates substantially impact battery life.

A third challenge we encountered is that many objects in the home, such as the telephone book and the first-aid kit, are seldom used (probably a couple of times a year). Therefore, to be useful, object usage sensors on these devices must have a very low false positive rate due to adjacent object usage or environmental noise (e.g. truck passing by). In our initial tests using simple acceleration thresholding to detect object usage, we found that random internal noise from the accelerometers, which were always sampling, would eventually (after days of use) trigger false positive activations when thresholds were set sufficiently low to capture real object usage.

Dealing with each of these three challenges impacts the fourth challenge: balancing good battery performance with robust object usage detection.

In this paper, we describe the practical issues we have encountered while trying to recognize object usage in real homes. We describe the advantages and disadvantages of existing sensing technologies used to recognize object usage, and present the design and evaluation of a specialized wireless sensor to detect object usage in existing homes. The sensor hardware and software was tuned based on our experience installing hundreds of sensors in real homes. Our two key design insights were to (1) minimize false positive activations due to adjacencies by implementing a classification rule tuned to differentiate real object usage from adjacent vibrations and noise in real-time and (2) extend battery life without sacrificing performance by combining two types of sensors with different strengths: a low-power piezofilm sensor with high sensitivity to external motion/vibration used for waking up a more energy costly accelerometer sampling at 20Hz used to differentiate between real and adjacent object usage. All technical specifications and microcode for the object usage devices developed in this work are available online [8].

2 Existing approaches to sense object usage

Previous ubiquitous computing and ambient intelligence studies where sensors have been installed in real homes for research (e.g., [1-3]) have often relied on the complex installation of reed switch sensors. A reed switch sensor typically installed in a volunteer's home consists of a data collection unit, a reed switch, and a magnet. All three components must be carefully aligned on an object (e.g. door, window or cabinet) in a way that properly activates the reed switch when the object is used. A single sensor of this type takes 5-10 minutes to install and test, and the installation of 200 of such sensors could require 16-32 man-hours of effort. This is a tremendous inconvenience to both the researchers and the subject in an experiment. The main advantage of using reed switches is good battery lifetime (possibly years, depending on object usage) that can be achieved even from small batteries since their associated circuitry can be kept in sleep mode until the reed switch is activated. However, reed switches only detect opening and closing events of objects such as doors, windows, cabinets, and drawers and are not well-suited for measuring other types of motion, such as the use of a table, chair, or appliance.

Object usage generally causes object movement or changes in object vibration, tilt, or acceleration. Therefore power efficient sensors such as vibration and tilt switches (e.g. mercury and ball switches) can be used to recognize object motion and to wake up external circuitry whenever vibration or tilt changes are detected. Unfortunately,

these sensors are often uniaxial and orientation dependent, and their sensitivity to vibration or tilt is set during manufacturing. The ability to modify the sensor's sensitivity is important for detecting object usage since household objects exhibit very different motion characteristics, as previously discussed.

Another alternative for measuring object usage in a power efficient way is to use a piezoelectric film sensor. The use of piezoelectric films as a low-cost and power efficient method to measure hand jerk motions (over 2.5G) in group interactions during collective events was proposed in [9]. Piezofilm sensors can be extremely sensitive to motion, detecting even tiny vibrations in the environment; however, they do not measure static acceleration (orientation with respect to ground) that is important for detecting slow tilt/rotation movements.

Accelerometers based on micro-electro-mechanical systems (MEMs) can also be used to detect object usage by measuring changes in object static and dynamic acceleration. MEMs accelerometers can be very small and self contained so that they can be placed on nearly any household object, and installation can require no more than simply throwing a sensor in a drawer or using putty (adhesive material) to stick it in to a cabinet door. No multi-point alignment is required, so sensors that can be simply stuck on objects quickly, reducing installation time to as little as 36s per sensor, as reported in [10]. Installing 200 single-point-of-contact sensors in a home may take as little as 2 man-hours of effort, a tolerable amount of time not only for researchers but even possibly for end-users. The main disadvantage of using MEMs accelerometers to detect object usage is their relatively high energy consumption. For example, continuous sampling of the popular ADXL202 biaxial accelerometer (~500μA@3V) at 10Hz using a 3mA micro-controller results in a theoretical battery life of only 46 days from a CR2032 coin cell battery (220mA). Newer versions of the accelerometer such as the triaxial ADXL330 (~400μA@3V) can also be used, however, improvements in battery life are minimal. An obvious way to increase battery life would be to increase battery size; however, this would make the sensor nodes larger and therefore less flexible and easier to dislodge.

A promising alternative to detect object usage is to use touch, or hand-to-object proximity. Battery-less passive radio frequency identification (RFID) tags are a cost effective way to measure object touch if a person is willing to wear a reader on or near the hands, embedded in bracelets or gloves [4, 6, 7]. RFID tags cost less than $1 US, and are small enough to be installed on tiny low-value objects such as office supplies. However, end users may be unwilling or unable to constantly wear a device on the hand. In addition, existing wearable RFID readers must be frequently recharged (e.g. every 2-10 hours [4, 11]) which could be potentially more annoying than replacing sensor batteries once or twice a year. Until wearable RFID readers are improved, passive RFID tags also require a good alignment and short distances (e.g. 10cm) to the wearable reader to be read [4, 6, 7]. The reader range could be extended, but this is at the expense of battery life and an increased number of false positives due to tags in the vicinity of objects being manipulated [4, 6, 7]. Passive RFIDs can also suffer from severe interference caused by metal objects and water-rich materials [12], sometimes negatively impacting performance on common objects such as food cans, silverware, door knobs, and liquid containers.

Technology	Assumptions	Advantages	Disadvantages
Magnetic reed switches	Object has a distinct open or closed state, with change indicating usage	Low power; relatively robust to adjacent object motion; relatively inexpensive $2-5 (compared to MEMs accelerometers)	Difficult to install (3 contact points); sensitive to alignment; only senses opening and closing events
Vibration and tilt switches	Object usage will result in object vibration or tilt.	Low power; self-contained single point of contact to object; easy to install. Wide price range depending on size and type ($1-10)	Sensitivity is orientation dependent (sensitivity for some orientations is zero); Cannot change sensitivity threshold; may trigger from adjacent vibration
Piezoelectric films	Object usage will result in object's dynamic acceleration change.	Low power; high sensitivity to external acceleration/vibration relatively inexpensive <$1.5 (compared to accelerometers); easy to install	Only senses dynamic acceleration (not static); may trigger from adjacent motion; usually uniaxial and with existing form factors, it would result in a cumbersome triaxial sensor since 3 films are required
MEMs accelerometers	Object usage will result in object acceleration (static or dynamic) change above accelerometer noise	Triggered by vibration, motion, and rotation of an object; single point of contact to object; easy to install; signal can be used to discern adjacent object motion	Power hungry (400-600μA@3V); relatively expensive ($7-12); internal sensor noise may result in false positives
RFIDs	Touch or proximity of object indicates use	Small; no battery; easy to install (a sticker); very inexpensive (<$1); single point of contact to object	Requires wearable or environmental reader; trade-off between reader's range and objects detected nearby (false positives); interference from metal or water; only indicates if object is present
WISP Intel	Object usage will result in object tilt	Single point of contact to object; easy to install; no battery; potentially same advantages as passive RFIDs and accelerometers	Requires environmental readers (~$1500 each); prototype stage (not yet extensively tested in homes); same disadvantages as RFIDs; susceptible to adjacent motion since simple thresholding is used to detect usage

Table 1: Advantages and disadvantages of existing sensing technologies to detect object-usage.

The wireless identification and sensing platform (WISP) [12] combines passive RFID tags and accelerometery. Readers distributed throughout a space constantly power the devices and motion sensing on the tags via a 1-bit accelerometer (mercury switch) is used to detect object usage. This approach is promising, since no battery replacement or wearable readers are required. However, the same challenges associated with passive RFIDs need to be overcome and since simple acceleration thresholding is used to detect object usage, false positives due to adjacent object manipulation remain a challenge. Consequently, the results presented in this work are also relevant in this setting.

Table 1 presents a summary of the assumptions, advantages, and disadvantages of the existing technologies available for sensing object usage.

3 Object usage sensor design

Our goals when creating the object usage sensor were to create a sensor that: (1) can be installed quickly by a non-expert, (2) can accurately detect object-usage in the wide variety of everyday objects with different motion patterns (vibration, tilt, rotation) and intensities without object specific thresholds, (3) minimizes the number of false positive activations due to adjacent motion and noise, and (4) extends the node's battery life by only processing the acceleration signal when meaningful motion is detected. In this section, we describe the hardware and software design inspired from our experiences installing hundreds of sensors in real homes.
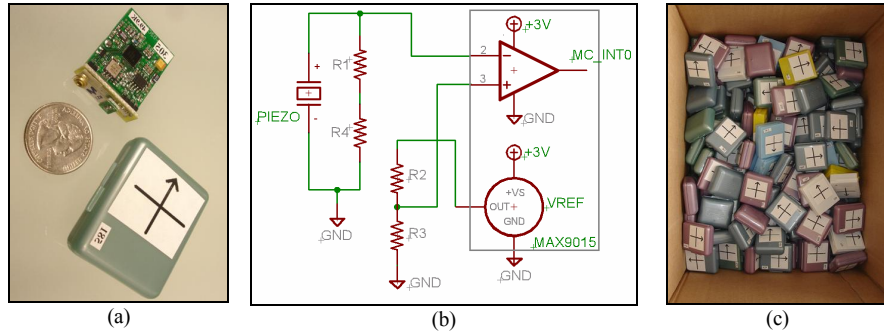
Figure 1. (a) Image of the object-usage sensor board and casing, (b) Schematic of the object-usage sensor wake-up circuitry, (c) Box of 243 sensors for collisions test.

3.1 Hardware design

The object usage sensor nodes developed in this work are built upon an existing wireless sensing infrastructure [10]. This sensing platform uses a featherweight MAC protocol in combination with a star network topology to save energy, simplify usage, and permit simultaneous reception from high and low sampling rate sensors. More details can be found in [10]. Even though the topology and communication protocols utilized in this platform are simple, they allowed us to easily test the performance of 168 newly developed object usage sensors in a real home (see Section 4.6). Moreover, the hardware and software principles on which the object usage sensors are based and presented in this work could also be used to improve sensor performance and battery life for sensor nodes using state-of-the-art mesh networking protocols.

The basic sensor board of the adopted wireless platform consists of a 3.2x2.5x0.6cm and 8.1g (including battery) stick-on wireless node that includes a 16Mhz 8051 microcontroller (μC), a wireless 2.4GHz transceiver, a 2-axis ±2G accelerometer (ADXL202, $12 US), and a MOSFET switch to power cycle the accelerometer from a CR2032 coin cell battery. Consequently, the new object usage sensor presented in this work is limited to this 2-axis accelerometer due to the unavailability of power efficient 3-axis accelerometers (e.g. ADXL330) at the time of development of the adopted sensing infrastructure. However, as described later, the 2-axis version is sufficient to achieve good performance in recognizing object usage in practice, although newer designs could utilize the new ADXL330 3-axis accelerometer without significant cost change. This basic sensor node was designed to recognize object usage by simply thresholding the 2-axis acceleration signal sampled at 10Hz against a predefined acceleration value [10]. The sensor broadcasts its ID whenever usage is detected. Using this simple thresholding approach to recognize object usage, the battery life of the sensor is about 46 days. This battery life may be sufficient for some ubiquitous computing test deployments, but the battery life is dependent upon the assumption that a sampling rate of 10Hz is sufficient to detect tasks with "instantaneous" stops such as shutting a drawer. When we deployed these sensors in a home, however, we ultimately determined that for some types of objects such as drawers and couches, the sampling rate of 10Hz was too low to consistently detect object usage
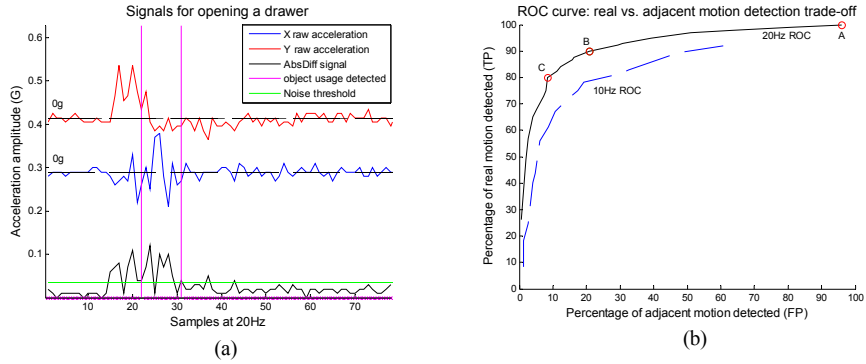
Figure 2. (a) Raw and preprocessed acceleration signals (XY AbsDiff) for an opening a drawer event sampled at 20Hz. (b) ROC curve showing the percentage of real vs. adjacent motion detected resulting from the brute-force search of the parameters at 20Hz and 10Hz.

when it occurred. We found that doubling the sampling rate to 20Hz improved object usage detection in these objects. Unfortunately, this new sampling rate reduces the battery life to less than a month. Therefore, we sought a way to improve the battery life of the sensor node. We chose to modify the hardware to use a sensor with high sensitivity to motion/vibration to wake up the 2-axis accelerometer and associated circuitry whenever meaningful external motion is detected for a short time interval. Consequently, energy use is preserved by keeping all circuitry in power down mode (2μA) until meaningful motion is detected by the wake-up sensor.

An obvious method to increase battery life without increasing the node complexity is to simply increase the battery size. For example, coin cells such as the CR2477 (10.5g) provide 1000mA of power, 4.5 times more power than the CR2032 used by the nodes. Nevertheless, this modification more than doubles the node weight and size, increasing installation complexity and probability of dislodgement, particularly in small objects. Furthermore, the better the sensor's power efficiency, the more useful the sensors will be in any application, no matter what battery type is used.

We considered three power-efficient sensor options to wake-up the circuitry: (1) tilt switches, (2) vibration switches, and (3) cantilever-beam piezoelectric films. Tests run to determine the sensors' sensitivity to external motion (see Section 4.1) indicated that the piezoelectric film was the best option to use. The piezoelectric film sensor (Minisense100 by MSI Inc.) works as a low-cost ($1.2 in qty of 50) accelerometer (1G) that measures dynamic acceleration without requiring external power. As a result, it can be used with auxiliary electronics to wake up the μC whenever the sensor node experiences an acceleration that exceeds a predefined fixed threshold V_{TH}. The hardware for the new object usage sensor board shown in Figure 1a consists of the basic sensor node board with an additional daughter board containing the circuitry shown in Figure 1b. As shown in Figure 1b, the output of the piezoelectric film is connected to the input of a nanopower comparator (MAX9015) that compares its generated voltage with a fixed voltage threshold V_{TH}. V_{TH} is obtained from a 1.236V voltage reference (MAX9015) and a voltage divider (R2, R3) to prevent variations in V_{TH} due to battery discharge over time. The output of the comparator is connected to

the μC external interrupt input to wake it up whenever the voltage generated by the piezoelectric film is greater than V_{TH}. V_{TH} was set in practice to maximize the sensor's response to external movement/vibration while preventing self-triggering by installing a variable resistor at R2 and R3. V_{TH} was finally set to 12.5mV, an acceleration threshold of 0.025g. The comparator and voltage reference (MAX9015) are always on but consume only 1μA. The production cost for the entire ad-on board (administration and NRE cost not included) is $3.9 US and was calculated assuming a production quantity of 50 and a two-month lead-time, as quoted by a U.S company (including PWB tooling, masks, stencils, soldering, and no electrical testing). This adds only 13% to the cost of the entire sensor board but substantially improves battery life. The cost of the entire sensor board (Figure 1a) under the same previous assumptions is $32.3 US. Note that 37% of the cost comes from the cost of the accelerometer ($12 US).

3.2 Object-usage detection algorithm

By adding the piezo-based wake-up circuit, the sampling rate can be increased without reducing battery life. However, adjacent motion still has to be differentiated from real motion to minimize false positive activations. The main purpose of the object-usage detection algorithm that runs on the sensor node is to recognize when an object is being used by analyzing the object's acceleration signal while minimizing false positives due to adjacent object manipulation and noise. Minimizing the number of false positives helps by (1) extending battery life because each wireless transmission consumes power, (2) simplifying raw data analysis and visualization for researchers (e.g. ethnographers), and (3) minimizing potential problems when pattern classification or machine learning algorithms are used to infer context from object usage.

The first step of the object usage detection algorithm consists of preprocessing the acceleration signals over each axis (X, Y) by computing the absolute value of the difference between adjacent samples to achieve orientation invariance.

$$AbsDiff = |current\_sample - last\_sample|$$

This step eliminates the DC component of the static acceleration that appears whenever the sensor is rotated with respect to ground. Orientation invariance is important because a person installing the sensors should not be required to orient them in a particular way, increasing installation complexity. Note that even though the DC signal component over each axis is eliminated, changes in rotation are still detected.

The second step consists of creating a new signal XY generated by selecting the preprocessed signal sample with the maximum value for both axes at each given instant in time (sample). This composite XY AbsDiff signal represents the overall motion experienced by the sensor in all axes. This step reduces complexity, since any processing performed on the accelerometer signals is performed only once over the composite signal instead of two times over each individual axis. Figure 2a shows an example of the 20Hz raw acceleration (X, Y) and the XY AbsDiff composite signal, and two object usage detections for an "opening a drawer" event. The third step is to differentiate meaningful acceleration samples due to true external motion from the oscillations generated by internal sensor noise. To achieve this, the maximum noise over the XY AbsDiff signal was measured from 3 sensor nodes sitting undisturbed

Drawer	Cabinet adjacent	Couch adjacent	Cup	Faucet
Drawer adjacent	Door	Chair	Remote Control	Bed/pillow
Cabinet	Couch	Table	Toilet cover	

Table 2. Fifty training examples were collected from each of these objects (or adjacent objects) to determine an algorithm that could discriminate real object usage motion from adjacent.

Feature	Description
Min value	Min value above noise level of the XY AbsDiff signal
Max value	Max value above noise level of the XY AbsDiff signal
Area under curve	Sum of all values of XY AbsDiff signal
Duration of motion	Number of samples greater than the noise level in XY AbsDiff signal
Mean	Mean of the AbsDiff signal computed sample-by-sample
Variance	Variance of the AbsDiff signal computed sample-by-sample

Table 3. Features extracted that can be computed efficiently by a low-cost low-processing power microcontroller.

on a table overnight (without the piezofilm add-on, so that the sensors were sampling the accelerometer continuously). The maximum amplitude, or worse case internal noise level found was 14 (35mG). Thus, when accelerometer readings above this threshold are obtained, the signal is assumed to be from actual motion.

In the forth step, real motion is differentiated from adjacent motion in real-time using the composite XY AbsDiff signal and a simple classification rule. To generate the classification rule, we collected 50 examples of the usage of each of the objects shown in Table 2 from a real home using a wired biaxial accelerometer sampling at 20Hz. In total we collected 550 examples of real object usage and 150 examples of adjacent object usage. The objects considered had a wide range of observed motion patterns, such as vibration (table, faucet), fast opening/closing motion (cabinets), smooth one-axis motion with rapid start/stops (drawers), rotation/tilt (toilet cover), extended motion (e.g. remote control) and combinations of these. After analyzing the adjacent object usage examples, we found that 96% (144) of them had XY AbsDiff signals greater than the noise threshold. Thus, if simple thresholding were used to detect object usage, 96% of adjacent object usage would be detected as real usage.

The features shown in Table 3 were computed over the XY AbsDiff signal of the segmented examples (max length of 4s). We used these features and the 700 examples to evaluate the performance of algorithms that generate classification rules implementable as *if-then* clauses. These *if-then* clauses and the features shown in Table 3 can be computed efficiently in a low-cost low-processing-power microcontroller. Obviously, more complex algorithms such as Hidden Markov Models (HMMs) that take into account temporal information could also be used. However, their computational requirements are too high to be implemented in most low-cost microcontrollers (<$5) currently available with real-time performance.

The WEKA toolkit [13] was used to evaluate the performance of the rule-generating algorithms shown in the first column of Table 4 over subsets of the features in Table 3 using 10-fold cross validation. The number of *if-then* clauses required to implement the resulting rules in C code was also defined as a complexity measure. Table 4 presents a summary of the performance results. The performance of the Naïve Bayes classifier was also investigated as a comparison baseline. Table 4 shows the results for the two single best discriminant features: *duration of motion* and *area under curve*. From Table 4, we observe that the rules generated by Ripper using only the duration of motion feature achieve the best compromise between classification

Algorithm	All features	Duration + Area + Variance	Duration	Area
Naïve Bayes Classifier	81.2 (NA)	83.1 (NA)	86.1 (NA)	77.0 (NA)
Rules PART	90.0 (21)	89.2 (9)	87.8 (3)	88.7 (3)
Ripper down rule learner (Ridor)	89.6 (12)	89.3 (4)	87.9 (5)	88.3 (7)
Ripper (JRip)	90.1 (8)	89.4 (8)	**88.7 (2)**	88. 9 (4)
C4.5 decision tree (J48)	90.3 (24)	89.6 (10)	87.8 (4)	89.0 (4)

Table 4. Classification performance using 10-fold crossvalidation over the 700 examples collected. The complexity of the classification rules produced is shown in parenthesis.

performance (88.7%) and complexity (2). The performance is slightly higher (~1.3%) when using all the features; however, the complexity involved in computing the features and implementing the decision rules increases substantially.

The classification rule generated by Ripper is: *If duration>7 then motion=real; Else motion=adjacent*. Using this simple rule, real motion is detected whenever 7 samples (at 20Hz) over the noise threshold are observed in the combined XY AbsDiff signal within a 2-4s window length (or length of usage example). To better understand the trade-off between *real* vs. *adjacent* detections (TPs and FPs), we performed a brute-force search over the parameter space (duration, window length) to generate an ROC curve. This was achieved by simultaneously varying the window length from 1 to 60 samples (.05s to 3s at 20Hz), varying the *duration of motion* threshold from 1 to 10 samples, and measuring the classification rule performance on the example set. Figure 2b presents the ROC curves (at 10 and 20Hz sampling rates) where each point in the plot represents a parameter setting. Not surprisingly, point A in Figure 2b (*duration*=1, window=1) is the most sensitive parameter setting offering near 100% detection accuracy for real motion but results in too many false positives for adjacent motion. The final parameters selected were *duration*=7, and window=29 (1.45s) or point B in Figure 2b. This corresponds to a real motion detection of 90% and adjacent motion detection of 21.5%. A lower FP detection of ~8% can be obtained using the settings of point C (*duration*=10, and window=29) but decreases real motion accuracy to ~80%. The ROC curves can be used during data collection deployments to trade detection performance with battery life depending on the application. Finally, these results were obtained by weighing the importance of all objects equally.

In summary, the object usage detection algorithm simply counts the number of times the XY AbsDiff acceleration signal samples gathered at 20Hz rise above the noise level threshold (35mG). If the count reaches 7 samples, the device "detects" real motion and will transmit a sensor ID to a nearby receiver. The count is reset each time 29 samples (1.45s) are observed without any samples above the threshold.

In field deployments, we noticed that it is not unusual for sensors to be placed on or near objects that move for very long periods of time continuously (e.g., a phone, running dishwasher, coffee maker, favorite chair, or leaking toilet). We therefore added a continuous usage/motion filter to minimize energy consumption whenever an object is moved for prolonged periods of time. From column 3 of Table 6 (discussed later), we observed that number of sensor firings for objects normally not used continuously was between 1 and 4. Consequently, the filter activates whenever the sensor has detected real object usage 4 consecutive times (within 1.4s of each other). While the filter is on, no ID broadcasts are made, and the filter turns itself off after no movement is detected for 2s. When the filter turns off, a special code is broadcast to the receiver.

Sensor	Main Axis	Drawer (%)			Cabinet (%)			Toaster (%)			Toilet (%)			Total (%)
		X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z	
Tilt	X	80	20	60	90	100	40	40	40	10	60	70	10	51.6
Vibration	Z	90	60	50	90	100	10	100	40	40	80	100	30	65.8
Piezofilm	X	90	80	80	90	100	80	70	60	70	70	90	70	79.1

Table 5. Sensitivity (%) of tilt, vibration, and piezoelectric film sensors to minimal external motion in all directions (X, Y, and Z) for four household objects when manipulated 10 times.

4 Evaluation

This section describes the tests performed during the development of the object usage sensor and its evaluation in a real home during realistic and worse case conditions.

4.1 Sensitivity to installation orientation

The sensitivity of a ball tilt switch (107-2006-EV by Mountain Switch), a miniature ball vibration switch (UBALL0100 by Particle Computer), and a piezoelectric film (Minisense100 by MSI Inc) were measured to determine their ability to rapidly wake-up the accelerometer and detect object usage accurately in any orientation (X, Y, and Z). All these sensors are uniaxial and therefore they become less sensitive as they are turned away from their main sensitivity axis (i.e. are sensitive to installation orientation). Ball-based tilt and vibration switches were preferred over mercury based switches, because they are usually more sensitive. This test was performed by comparing three sensor nodes sampling at 20 Hz each using a different sensor type as wake-up option and one sensor without the wake-up circuit (accelerometer continuously powered up). All used simple acceleration thresholding to detect object usage. These sensor nodes were placed in a drawer and cabinet (motion), a toilet cover (tilt), and on a toaster (vibration). Each object was manipulated 10 times in a worse-case scenario of minimal slow motion in each orientation (X, Y, and Z): the drawer and cabinet were opened, the toilet seat was raised, and the toaster activated. The results of comparing each sensor type are shown in Table 5. Note that wireless signal loss and lack of synchronization between the acceleration sampling between the sensors have a slight influence on the results. Overall, the piezofilm is the sensor with highest sensitivity to external motion in any direction. Consequently, only one piezofilm is required to act as a wake-up sensor. Orientation sensitivity problems with the tilt and vibration sensors could be overcome by using three sensors oriented along the X, Y, and Z axes. However, at current costs for the vibration switch at low quantities ($10 each), this solution would double the cost of the entire sensor node (~$59). For the tilt switch, its current size (16x4x4mm) would result in a sensor node with a cumbersome form factor that would increase installation complexity.

4.2 False positives from internal sensor noise

Recognizing object usage by simple thresholding of acceleration generates a large number of false positives due to (1) variations in the accelerometer internal noise amplitude due to fluctuations in the battery voltage while transmitting and (2) errors generated while reading the accelerometer values. Even when accelerometer values

are read reliably 99% of the time, sampling over a day sometimes generates sufficient erroneous readings to produce several false activations. In the new object usage sensor, the problem is addressed by only sampling the acceleration when necessary (when external motion is detected). To evaluate the FP activations generated due to internal sensor noise, 12 object usage sensors were left sitting undisturbed over a table in an office space for 24 days. The number of false positives found was zero.

4.3 Wireless link performance and percentage of collisions

To evaluate the performance of the object usage sensor wireless link to interference caused by nearby metal objects and surfaces (a problem in RFIDs), we installed the sensors on a metal tuna can, a metal doorknob, and a metal spoon. A computer with a receiver was placed a few meters away. The objects were picked up and put down or used normally 10 times each. All 30 of the usages were detected. The objects were placed on many metal objects in the realistic home test reported shortly.

A concern when using simple MAC protocols and a star network topology, such as the ones used by the sensor nodes, is collisions. Therefore, we placed 243 object usage sensors in a large box, as shown in Figure 1c to measure collisions due to simultaneous activation of sensors in a real-world worse-case scenario. A computer with a receiver was placed a few meters away. The box was kicked hard 7 times (with a few seconds of no motion in between each kick) and then moved from one part of a room to another (taking exactly 5s per movement) five times. Each of these actions is very likely to activate each of the sensors in the box simultaneously. In practice, 33.7% (82) of the sensors wireless ID transmissions were detected in the kick experiment (the 66.3% not detected were lost due to collisions). We think that it is unlikely in a home environment that any typical "instantaneous" activity will result in 80+ sensors firing at the exact moment. In the 5s carry event, only 26.4% of the sensors transmissions were lost due to collisions. Consequently, the simple MAC protocol and network topology utilized is capable of detecting the simultaneous transmission of a large number of sensors and appropriate for the tests performed in this work. Collisions could be further minimized using a more sophisticated wireless protocol, as the expense of possibly requiring more costly electronics per sensor node. More details on theoretical collision tests based on number of nodes, transmission rate and time can be found in [10].

4.4 Battery life

The battery life of the object usage sensors is dependent upon the usage of the object on which the sensor is placed. At one extreme is a sensor that is never moved. The idle theoretical battery lifetime of our new object usage sensor when it undergoes no motion and consumes 4μA in sleep mode is approximately 6.3 years from a CR2032 coin battery (220mAh).). In practice, we tested the idle battery life (self discharge) of 12 sensors by leaving them sitting undisturbed on a table in an office space for 9.5months. The sensors were still functional and with approximately 80% battery charge after the end of this time period. This gives us a more realistic idle battery life upper-bound estimate of roughly 4 years. At the other extreme is a sensor that is moving continuously, where the extended movement filter is disabled. In that condition,

Object	ACC	TP	FN	FP	CF	Brief explanation (*)
Cabinet (open)	100	30 (1.8)	0	0	0	-
Cabinet (close)	100	30 (2.2)	0	2*	0	Cabinets slammed 2 times
Drawer (open)	96.6	29 (1.3)	1*	0	0	Drawer opened slowly
Drawer (close)	100	30 (1.8)	0	3*	0	Drawer slammed 3 times
Front door (open)	100	30 (3.3)	0	0	0	-
Front door (close)	100	30 (2.4)	0	3*	0	Door slammed 3 times
Wood chair on wood floor (sit down)	100	30 (2.3)	0	0	0	-
Wood chair on wood floor (get up)	100	30 (2.8)	0	1*	0	Table moved while sitting
Upholstered couch (sit down)	86.6	26 (1.2)	4	5*	0	Adjacent cushions activated
Upholstered couch (get up)	63.3	19 (1.1)	11*	2	0	Less motion than sit down
Coffee mug (sip drink)	100	30 (4.9)	0	1*	0	Dropping mug on table
Computer keyboard (typing)	100	30 (6.8)	0	0	18	-
Electric can opener (open a can)	100	30 (4.9)	0	3*	27	Spice rack beside can opener
Remote control (change channel)	100	30 (5.4)	0	0	24	-
Table (put full mug down)	20.0	6 (1)	24*	0	0	Similar to adjacent motion
Table (sign paper)	53.3	16 (1)	14*	0	0	Not enough vibration/motion

Table 6. Accuracy (ACC) [TP/(TP+FN)], TPs, FNs, and FPs (of nearby sensors) in a controlled test where 3 subjects used 10 home objects 30 times each. The number of average activations reported during each object usage is shown in parenthesis. The column CF shows the number of motion events for which the continuous motion filter was activated.

the sensor will sample from the accelerometer and transmit continuously. Three such sensors were installed on a rotating arm. The battery was measured to last 7 days and broadcast a total of 15.4 million sensor activations. If the motion filter is on, the sensors can run continuously for approximately one month and a half because the device is not transmitting continuously. To estimate the typical lifetime of a sensor installed in a home, we assume that the sensor activates 20 times a day for 3s (average motion duration for opening/closing a drawer). In that case, based up on the measured power consumption of the accelerometers and wireless transmission, the node's battery life is estimated to be 1.1 years.

4.5 False positives from movement of nearby objects

As part of a controlled test, three subjects were asked to each use 10 home objects (with sensors attached) normally 30 times. Table 6 shows the true positive (TP), false negative (FN), and false positive (FP) activations for the total of 480 object usages. The FPs correspond to sensor firings of nearby objects. The table also indicates, in parenthesis, the average number of activations reported by the sensor during each object interaction. From this number, we can see that opening events in cabinets and drawers are more difficult to recognize than closing events. Interactions that take longer clearly generate more activations, such as use of the keyboard. We can also observe that the number of FNs is near zero for most objects (8.9% overall) except for extremely low magnitude usage events such as signing a paper on a table (sensor under table) and getting up from a couch (sensor underneath center cushion). Getting up from the couch is also difficult to recognize because of wireless signal loss due to body blocking. Putting down a mug on a table was difficult to recognize because it generates short duration signals similar to the ones generated by adjacencies. Finally, the number of FPs is high for worse-case conditions such as slamming events or inevitable adjacent motion. For example, sitting on the center cushion of a three-seat couch induced significant motion on the two adjacent cushions. FPs are also slightly

higher by design since we decided to err on the side of being too sensitive rather than not sensitive enough, given that higher level inference algorithms may be able to further filter FPs. This relatively controlled test suggests that the sensors are able to detect object usage well in a variety of everyday objects.

4.6 Realistic performance in an instrumented residential home

We wanted to be certain that the sensors would function well when deployed in a real home when people use objects normally. To test this, we used an instrumented condominium that has hard-wired reed switches in all the cabinetry (cabinets, drawers, doors, and appliance doors) and a comprehensive audio-visual recording system for analysis of activity and sensor firings by researchers [14].

As a first test, two volunteers (a young married couple) not affiliated with our research group were recruited and asked to live at the condominium for a period of 2.5 months. The volunteers performed their normal daily activities while video was being recorded and 168 object usage sensors were attached to objects in the environment. Sixty-six object usage sensors had hard-wired reed switch equivalents and 102 were installed on objects that had no wired equivalent (e.g. sofa, appliances, chairs, etc).

The second month of data were used to compare the wired switch sensor activations to the wireless sensor activations. We experimentally confirmed that the detection accuracy for a wired switch sensor is 99% for opening/closing events by opening and closing every cabinet and drawer at the condominium ten times. For the remainder of this analysis, we assume the wired switches accurately detect open/close events.

The overall accuracy for detecting opening-closing events of the wireless sensors with respect to the wired switches was 86.6%±16.0. An object usage sensor was considered to correctly detect a wired switch opening-closing event whenever it fired within ±5s of the wired switch activation. This ±5s window was selected to permit a temporal comparison due to delays observed in practice in the switch sensor network. After building a 1s resolution histogram over the activation time differences between wired and wireless sensors over the ±5s windows, we determined that 77.8% of the activations fire within ±1s of each other, 91.4% within ±2s, 95.8% within ±3 and 99.3% within ±3s. Delays greater than +2s represent cases where the object usage sensor does not fire instantly when an object is used (e.g. a cabinet door is opened), but instead fires sometimes during the fluid motion. Negative delays indicate that the wireless sensor fired before the wired switch due to delays in the wired sensor networks (worse case is 2s).

There were a total of 124,064 total sensor activations where 12,099 were generated by the wired switch sensors and 111,965 by the object usage sensors. Note that there are more object usage activations because each wired switch event generates varying numbers of wireless sensor activations (e.g. 1-4) depending on the object type, as shown in parenthesis in column 3 of Table 6. Table 7 summarizes the comparison results for the three object types (cabinets, drawers and doors) that had equivalent wired switches. The table also shows the performance for some individual objects (results located at the bottom of Table 7 are worse-case scenarios). The TPs shown in the table are the number of detected wired switch activations, FNs are the wired

Overall summary statistics				
Object	**TP (μ ± σ)**	**FN (μ ± σ)**	**Other**	**Brief explanation (*)**
Cabinets	85.6 ± 17.6	14.3 ± 17.6	53,323*	Doors swung but not closed
Drawers	87.5 ± 12.3	12.4 ± 12.3	1,037*	Drawer contents manipulated
Doors	93.6 ± 4.0	6.3 ± 4.0	33,341*	Dishwasher motor vibration
Example results for some individual objects from good (top) to worse (bottom) performance				
Object (sensor ID)	**TP (%)**	**FN (%)**	**Other**	**Brief explanation (*)**
Cabinet door hallway pantry (738)	36 (100)	0 (0)	57*	Cabinet's content manipulated
Cabinet door dining room (1009)	44 (97.7)	1 (2.3)	36*	Cabinet's content manipulated
Drawer office desk (767)	32 (96.9)	1 (3.1)	57	Drawer's content manipulated
Cabinet door master bathroom (611)	27 (87)	4 (13)	47	Cabinet's content manipulated
Drawer office desk (762)	12 (70.5)	5 (29.5)	23*	Drawer's content manipulated
Drawer utensils kitchen (530)	81 (72.9)	30 (27.1)	95*	Drawer's content manipulated
Cabinet door hallway pantry (979)	94 (80.3)	23 (19.7)	106*	Vibration induced by washer
Drawer kitchen (963)	45 (95.7)	2 (4.3)	247*	Cabinet's content manipulated
Master bathroom door (623)	700 (97.7)	16 (2.3)	2,004*	Door swung but not closed
Apartment front door (594)	507 (90)	52 (10)	1,249*	Hydraulic holder at door
Dishwasher door (792)	180 (90)	18 (10)	29,030*	Dishwasher motor vibration
Refrigerator door (996)	12 (20)	47 (80)*	143	Wireless signal loss on metal

Table 7. Summary of the comparison results for all the three object types (cabinets, drawers and doors) that had equivalent wired switch sensors and some particular examples of the two objects with best and worse performance. True positives (TP), false negatives (FN), and other object usage activations are also shown.

Object	Activations	Object	Activations
Computer keyboard	7,765	Wood chair	321
Remote control	3,519	Toilet flush	301
Upholstered couch	451	Toilet dispenser	255
Table	587	Can opener	68

Table 8. Examples of the number of object usage sensor activations during one month for some objects that did not have a wired switch equivalent.

switch activations not detected, and 'other' is the number of times the object usage sensors fired when no equivalent wired switch activation was found. This happens while manipulating the contents of drawers and cabinets after they have been opened, or when doors are swung without being fully opened or closed (e.g. holding door to let someone in). Note that 'Other' activations are not directly comparable to TPs and TNs since reed switches generate one activation per opening/closing event and object usage sensors generate multiple. The dishwasher door (as show in Table 7) is an example when the number of 'other' activations is high because intermittent vibration of the motor constantly triggers the object usage sensor while the wired switch is only activated during the dishwasher door opening and closing. It is unfair to consider these activations as FPs since vibration is induced at the sensor by real object usage. On the other hand, the refrigerator door is an example of a high number of FNs produced because of wireless signal loss due to a large metal surface area on the refrigerator's door. Note that this is the only case where we experienced problems due to surrounding metal surfaces/objects Table 8 shows examples of the number of activations for object usage sensors that did not have wired switch equivalents over one month of data. For some objects that are continuously manipulated, such as the remote control and the computer keyboard, the number of activations is high, despite the use of the continuous motion filter. Among the most activated sensors we find the master bathroom door, the half bathroom door, the dishwasher, computer keyboard,

laptop, remote control, and kitchen faucet. Among the least activated sensors we find the backyard sliding door, coffee table and some kitchen cabinets and drawers.

Finally, during the 2.5 month data collection, 23 of the most used sensors (e.g. remote control, computer keyboard, laptop, telephone, office chair, toilet flush, eyeglasses case, and coffee mug) required battery replacement after two months. The battery drain results because the continuous filter only saves energy by preventing wireless broadcasts but still wastes energy by keeping the accelerometer on. This problem and our decision to err on the side of sensitivity are the reason why the battery life observed in practice for heavily used objects differs from the battery life estimated for a typical object in the home (1.1 years). Future versions of the continuous filter might reduce the sampling rate of the accelerometer dynamically when continuous usage is detected to further improve battery life. In addition, future versions could increase the sample rate of the accelerometer after recognition of the first activation to raise the detection ratio and noise rejection at the expense of battery life.

From the previous evaluation comparing wired switches and wireless object usage sensors, it was clear that we required another test that allowed us to evaluate the performance of wireless sensors with no wired switch equivalents in more detail. Consequently, we performed a second test where a person busily cooks a pancake recipe and cleans the house (sweeping, vacuuming, doing laundry, doing dishes, wiping a surface, etc.) for a period of four hours while video was being recorded. These activities can be considered a worse case scenario to measure object usage because the kitchen has the largest sensor density and a large number of objects are manipulated or accidentally bumped into during cleaning activities (e.g. putting away things, wiping a surface, and mopping). The person was affiliated with our research group but not knowledgeable about the technical operation of the object usage sensors. A trained annotator (meticulously) labeled all object usage events (object touches >2s) from the four-hour dataset and video. We then computed the TP, FN, and FP detection rates for each object that had an object usage sensor placed on it. A sensor was marked as correctly detecting an object usage event if it activated at least once when the object was being manipulated. For example, a drawer manipulation that is a quick open and shut would be counted as one movement while manipulation consisting of an open, release, pause, and then grab and close would be counted as two usage events. The number of FP activations induced in sensors by the usage of objects nearby the object being used was also computed.

Over the course of the four hours, the annotator found 327 distinct object manipulations. The wireless sensors detected 276 of the 327 object manipulations giving a recognition accuracy of 84.4%. The number of FNs found was 51 (15.6%). Table 9 shows the break down of the FNs and FPs and provides a brief explanation for their generation. Note that the FPs shown in Table 9 are given in sensor activations while FNs are given in object usages. As a result, since each object usage generates several (1-7) sensor activations depending on object type, it is not possible to directly compare FPs and FNs. From Table 8, we can observe again that most FNs generated are either due to very slow motions or wireless signal loss due to metal surface on fridge's door. We can also see that the main sources of FPs are worse-case scenarios of vibration traveling through surfaces such as slamming events, constant vibration in kitchen island due to usage of a 12kg mixing machine, and vibration traveling

	Num	% FN	Reason for generation
FN	17	33.3	Sink faucets not fired while opened or closed slowly
	15	29.4	Wireless signal loss in 2 sensors due to large metallic surface of refrigerator's doors
	7	13.7	Drawers and cabinets misses due to slow opening/closing events
	6	11.7	Checking on meal status events where oven's door was not fully opened or closed
	5	9.8	Swinging doors without closing them completely
	1	1.9	Touch of upper part of rotating chair with no movement of base (sensor location)

	Num	% FP	Reason for generation
FP	121	46.8	Industrial power mixing machine (KitchenAid 325W, 12.6kg) on at kitchen island
	56	21.7	Vibration propagated to objects located on top of sink (e.g. dishwashing liquid, liquid hand soap, and faucets) through metal surface while washing dishes
	35	13.5	Slamming drawers and cabinets (including kicking drawers to close them)
	34	13.1	Dropping heavy objects (flour, mixer, stove burners) on surfaces
	4	1.5	Bumping objects unintentionally while cleaning a surface or manipulating an object
	4	1.5	Bumping sensors installed at cabinetry (bottom) while vacuuming and mopping
	2	0.7	Kicking sensors with feet while standing in front of cabinets or drawers
	2	0.7	Firing of sensors located at the bottom drawer inside refrigerator when refrigerator turned on cooling mechanism

Table 9. Explanation of FNs (given in object usages) and FPs (given in sensor activations) in the four hour test. The % column shows the percentage with respect to the total number of FNs and FPs.

through sink's metal surface to objects on the sink while washing dishes. This test shows how surprisingly difficult is to create an object usage sensor that works in a real environment. Application designers must be wary, for example, of using single activations to trigger events. That said, we think these results show that our sensors provide a nice compromise: good battery life and good object-usage detection with what may be an acceptable level of FNs and FPs for real-world situations. Furthermore, a portion of the FPs generated result from our decision to err on the side of higher sensitivity. FPs could be reduced by using other parameter setting from the ROC curve (Figure 2b) at the expense of lower real object usage recognition.

An interesting path for future research might be to combine proximity sensing [4, 6, 7] and motion sensing to partially mitigate the adjacency problem in some applications. In this approach, hand proximity to an object could be measured by the use of a wearable technology (e.g. RFID enabled wristwatch) and direct manipulation by object movement.

6 Conclusions

In this work we have described the challenges and practical issues involved in recognizing object usage from object motion in everyday environments -- a problem that appears trivial at first glance but turns out to be surprisingly difficult. We have presented a portable wireless sensor to detect object usage in existing homes that (1) is easy to use and easy to install, (2) extends battery life for longitudinal deployments by combining two sensing modalities, and (3) minimizes false positives due to adjacent movement of objects and environmental noise. The sensor parameters suggested err on the side of sensitivity to object usage to minimize false negatives because offline filtering could further reduce false positives. However, the sensor parameters can be modified allowing researchers to trade real vs. adjacent object usage recognition accuracy and battery life depending on the application and duration of sensor de-

ployments. We evaluated the object usage sensors in a non-laboratory setting and showed that they can detect object usage with an accuracy of 86.6% when compared with wired reed switches over one month of data and with an accuracy of 84.4% in difficult scenario where a person is busily cooking and cleaning in a cluttered kitchen. Finally, we also characterized the situations where the sensors have failed and suggested possible paths of future improvement.

References

[1] E. Munguia Tapia, S. S. Intille, and K. Larson, "Activity Recognition in the Home Setting Using Simple and Ubiquitous Sensors," in *Proceedings of PERVASIVE 2004*, vol. LNCS 300. Berlin: Springer-Verlag, 2004, pp. 158-175.

[2] D. Wilson, "Simultaneous Tracking and Activity Recognition (STAR) Using Many Anonymous, Binary Sensors," in *Proceedings of The 3rd International Conference on Pervasive Computing (Pervasive '05)*. Berlin: Springer-Verlag, 2005, pp. 62-83.

[3] T. Barger, D. Brown, and M. Alwan, "Health Status Monitoring Through Analysis of Behavioral Patterns," in *Proceedings of IEEE Transactions on Systems, Man and Cybernetics Part A*, vol. 35, 2005, pp. 22-27.

[4] M. Philipose, K. P. Fishkin, M. Perkowitz, D. J. Patterson, D. Hahnel, D. Fox, and H. Kautz, "Inferring Activities from Interactions with Objects," *IEEE Pervasive Computing Magazine*, vol. 3, 2004, pp. 50-57.

[5] J. S. Beaudin, "From Personal Experience to Design: Externalizing the Homeowner's Needs and Assessment Process," M.S. Thesis. Cambridge, MA: MIT, The Media Laboratory, 2003.

[6] M. Philipose, K. Fishkin, D. Fox, H. Kautz, D. Patterson, and M. Perkowitz, "Guide: Towards Understanding Daily Life Via Auto-Identification and Statistical Analysis," in *Proceedings of UbiHealth '03*, 2003.

[7] K. Fishkin, M. Philipose, and A. Rea, "Hands-On RFID: Wireless Wearables for Detecting Use of Objects," in *Proceedings of The Ninth Annual IEEE International Symposium on Wearable Computers (ISWC '05)*: IEEE Press, 2005, pp. 38-41.

[8] E. Munguia Tapia and S. S. Intille, "Environmental Sensors Hardware and Software Resources," [cited August 28th, 2007]. Available from http://architecture.mit.edu/house_n/MITes.

[9] M. Feldmeier and J. A. Paradiso, "Giveaway Wireless Sensors for Large-Group Interaction," in *Proceedings of The ACM Conference on Human Factors and Computing Systems (CHI '04)*: ACM Press, 2004, pp. 1291-1292.

[10] E. Munguia Tapia, S. S. Intille, L. Lopez, and K. Larson, "The Design of a Portable Kit of Wireless Sensors for Naturalistic Data Collection," in *Proceedings of PERVASIVE 2006,* B. Heidelberg: Springer-Verlag, 2006, pp. 117-134.

[11] M. Philipose, "Large-Scale Human Activity Recognition Using Ultra-Dense Sensing," The Bridge, National Academy of Engineering, vol. 35, no. 4, 2005.

[12] M. Philipose, J. R. Smith, B. Jiang, A. Mamishev, S. Roy, and K. Sundara-Rajan, "Battery-Free Wireless Identification and Sensing," *IEEE Pervasive Computing*, vol. 4, 2005, pp. 37-45.

[13] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd Ed: Morgan Kaufmann, 2005.

[14] S. S. Intille, K. Larson, E. Munguia Tapia, J. S. Beaudin, P. Kaushik, J. Nawyn, and R. Rockinson, "Using a Live-In Laboratory for Ubiquitous Computing Research," in *Proceedings of PERVASIVE 2006*. Berlin: Springer-Verlag, 2006, pp. 349-365.