Mimic Sensors: Battery-shaped Sensor Node for Detecting Electrical Events of Handheld Devices

Takuya Maekawa[1], Yasue Kishino[2], Yutaka Yanagisawa[2], and Yasushi Sakurai[2]

[1] Graduate School of Information Science and Technology
Osaka University
takuya.maekawa@acm.org
[2] NTT Communication Science Laboratories
surname.name@lab.ntt.co.jp

Abstract. In this paper we propose and implement a battery-shaped sensor node that can monitor the use of an electrical device into which it is inserted by sensing the electrical current passing through the device. We live surrounded by large numbers of electrical devices and frequently use them in our daily lives, and so we can estimate high-level daily activities by recognizing their use. Therefore, many ubiquitous and wearable sensing studies have attempted to recognize the use of electrical devices by attaching sensor nodes to the devices directly or by attaching multiple sensors to a user. With our node, we can easily monitor the use of an electrical device simply by inserting the node into the battery case of the device. We also propose a method that automatically identifies into which electrical device the sensor node is inserted and recognizes electrical events related to the device by analyzing the current sensor data. We evaluated our method by using sensor data obtained from three real houses and achieved very high identification and recognition accuracies.

Key words: Sensors; Electrical devices; Battery

1 Introduction

Daily activity recognition is one of the most important tasks in pervasive computing applications because it has a wide range of uses in, for example, supporting the care of the elderly, lifelogging, and home automation [15, 14]. Many studies have employed body-worn accelerometers to recognize human activities [10, 1, 12]. While the wearable sensing approach can sense users' activities in both indoor and outdoor environments, wearing sensor devices on parts of the body may place a large burden on the user's daily life. Many other studies have focused on sensors that monitor indoor environments, and have tried to recognize activities based on dense ubiquitous sensors such as RFID tags and switch sensors installed in the environments [17, 21, 22]. While the approach does not require the user to wear sensors, the costs involved in deploying and maintaining ubiquitous sensors are huge. Also, sensor nodes attached to many daily objects can detract from the aesthetics of artifacts in the home [2]. In addition, many activity recognition studies that employ machine learning techniques require the end-user to prepare labeled training data herself in her daily life environment.

As outlined above, many existing activity recognition approaches place different kinds of large burdens on users. We summarize the users' costs below.
- **Cost of deployment**: A user has to install many sensor nodes in her daily environment when she uses ubiquitous sensors. In some cases, the user should manually associate a sensor node with the daily object to which the sensor is attached. Before the deployment phase, no sensor node knows to which daily object it is attached.
- **Cost of maintenance**: In many cases, a user must regularly replace the battery in a sensor node. The user must also replace broken sensor nodes.
- **Cost related to long-term daily use**: A user must wear multiple sensor nodes in her daily life and/or use daily objects to which sensor nodes are attached. The sensor nodes can also detract from the aesthetics of the home.
- **Cost related to supervised machine learning**: Many ubiquitous sensing approaches and some wearable sensing approaches require labeled training data created in an end user's environment.

On the other hand, recently, many studies in the pervasive computing research field have attempted to monitor the usage of home electrical devices. Because we live surrounded by large numbers of electrical devices, and frequently use them when we perform daily activities, we can estimate high-level daily activities by recognizing the use of these devices. In addition, due to the growing interest in energy conservation, many studies have tried to monitor the energy consumed by electrical devices. Ubiquitous and wearable sensing approaches have been employed to detect the use of electrical devices. However, many existing approaches place large burdens on users as mentioned above. In this paper, we define a new sensing framework called the *mimic sensor framework* that does not impose large burdens on users. A sensor node designed based on the mimic sensor framework works unobtrusively by mimicking objects and plugs with standardized forms. An example sensor node based on the mimic sensor framework has the shape of objects with standardized forms such as an AA battery and an SD memory card and provides the functions of the original object, e.g., discharge and data storage functions. Because a user can use the sensor node in the same way as the original object, she can easily install the sensor node simply by inserting it into the battery case or SD card slot of an electrical device that she wants to monitor. The sensor node senses phenomena related to the electrical device and wirelessly sends the sensor data to a host computer. As described above, because a sensor node designed based on the mimic sensor framework has the shape of an object that exists in a user's ordinary daily life and the user can employ the node in the same way as the original object, we can monitor the user's daily life transparently.

In this paper, we implement a prototype battery-shaped sensor node as an example mimic sensor. The prototype node includes a battery and provides a current discharge function in the same way as conventional batteries. The node also monitors (senses) an electrical current that flows through the node when the node is inserted into an electrical device and then sends the sensor data to a host computer. We analyze the sensor data and recognize electrical events related to the electrical device. With a digital camera, for example, by analyzing the data we can recognize when the user turns it on and when she takes a picture. In addition, to eliminate the cost related to the association of sensor nodes, we try to automatically identify into which electrical device the sensor node is inserted by analyzing the sensor data.

In the rest of this paper, we first introduce work related to detecting the use of electrical devices, and then describe our definition of the mimic sensor framework. After that, we describe the design and implementation of our prototype battery-shaped sensor node. We also propose a machine learning-based method that identifies which electrical device a battery-shaped sensor node is in and that recognizes electrical events related to the device. The contributions of this paper are that we propose and develop a new battery-shaped sensor node, and propose a device identification and event recognition method by analyzing its data. We also evaluate our method by using sensor data obtained from three real houses.

2 Related Work

As mentioned in section 1, various ubiquitous and wearable sensing approaches have been proposed for recognizing human activities. Many ubiquitous sensing approaches employ a large number of small sensors such as switch sensors, RFID tags, and accelerometers attached to daily objects [22, 17, 20]. By using ubiquitous sensors, we can detect the use of electrical devices in addition to the use of daily objects. A system proposed in [7] employs ubiquitous sensor nodes equipped with magnetic sensors or light sensors attached to each electrical device to detect its use. Although the ubiquitous sensing approach can achieve fine-grained measurements of daily lives, its deployment and maintenance costs, e.g., battery replacement costs, are very high. Several studies have employed small numbers of sensor devices that monitor home infrastructures to detect the use of electricity, water, or gas in home environments [3, 4, 16]. In particular, the systems proposed in [16, 5] recognize the use of electrical devices by monitoring noise on home electrical systems. The systems focus on stationary electrical devices connected to home electrical systems via electric plugs. On the other hand, the battery-shaped sensor nodes proposed here are designed for use with portable electrical devices that are not connected to home electrical systems. The studies that come closest to our concept involve sensor nodes shaped like a power strip [8, 6]. The power strip sensor node has electrical outlets and supplies electrical devices connected to the outlets with electrical power. The sensor node also monitors electrical current drawn from each outlet. Because a user can employ the sensor node in the same way as a normal power strip, her daily activities can be monitored transparently. We consider the sensor node to be one example of a mimic sensor. As another similar example, we can assume a sensor node shaped like a USB hub. The sensor node has USB ports and monitors the use of electrical devices that are connected to the node such as I/O and data storage devices. Here, we consider that the above approaches and our approach are complementary rather than competing techniques because the above approaches can recognize the use of stationary electrical devices that run without batteries.

Most wearable sensing approaches use multiple sensor nodes attached to the wearer's body [1, 19, 13]. Although these approaches can recognize the wearer's activities in outdoor environments, they impose the burden of the need to wear several sensors during daily life. The system proposed in [11] recognizes the use of portable electrical devices held by a user by employing several magnetic sensors attached to her hands. The system captures magnetic fields emitted by magnetic components such as coils and permanent magnets in portable electrical devices and identifies which electrical device the user is using.

3 Mimic Sensor Framework

As mentioned in section 1, a sensor node designed based on the mimic sensor framework has a standardized form or a plug with a standardized form, which means that the sensor node can be connected to another device or a socket. The sensor node basically receives electrical power via the connection. The sensor node also senses data related to the device or socket to which the node is connected. We consider that sensing the electrical current that flows through the node via the connection may be useful and effective for detecting electrical events related to the device. Note that the sensor node can include other sensors such as an accelerometer and a temperature sensor. This permits us to capture additional information related to the device usage. Examples of sensor nodes based on the mimic sensor framework include an AA battery-shaped sensor node, an SD card-shaped sensor node, a flash memory card-shaped sensor node, a light bulb-shaped sensor node, a fluorescent light-shaped sensor node, a power strap-shaped sensor node, and a USB hub-shaped sensor node. The sensor nodes provide the functions of the original objects that they are mimicking. Therefore, the user can employ the sensor node in the same way as the original object.

In section 1, we mentioned four kinds of burdens placed on users. Here we explain how the mimic sensor approach reduces these burdens.

- **Cost of deployment**: Some studies employ such ubiquitous sensors as RFID tags, infrared sensors, and switch sensors [17, 9, 21]. However, deploying such sensors requires the user to have specialized knowledge. On the other hand, because the user can use mimic sensor nodes in the same way as the original objects, she does not require specialized knowledge.
- **Cost of maintenance**: Because mimic sensor nodes basically receive electrical power from other devices or sockets to which they are connected, a user need not replace their batteries. Note that the user must recharge/replace the battery included in a battery-shaped sensor node when it runs out. However, this is also the case even if the user employs a regular battery in place of the node. Therefore, the battery-shaped sensor node does not impose any additional burdens related to battery replacement. Note that when the sensor node does not have a power-saving architecture, the battery replacement interval is shortened.
- **Cost related to long-term daily use**: Because such ubiquitous sensors as RFID tags and switch sensors are usually attached to daily objects, they detract from the aesthetics of those objects. On the other hand, a user can use mimic

sensor nodes exactly as she uses the original objects and so her life remains unchanged by sensor installation. Moreover, since such sensor nodes as battery-shaped sensor nodes and SD card-shaped sensor nodes are inserted into electrical devices, the user is not aware of them.
- **Cost related to supervised machine learning**: Many ubiquitous sensing approaches require labeled training data obtained in each user's environment to recognize activities in that environment. However, such data is very costly to prepare. We assume that we sense the amount of electrical current that flows through a mimic sensor node when the node is connected to a device. We consider that the flow characteristics are device-dependent and so users can share training data obtained from the device. Assume that a battery-shaped sensor node is inserted into a CD player in a house, and the player plays music. Sensor data (time-series current values) obtained from the player may have a characteristic frequency and the same model CD player in another house may also have the same characteristic frequency. Therefore, end users need not prepare training data in their houses.

4 Prototype Battery-shaped Sensor Node

We design and implement a prototype battery-shaped sensor node as an example mimic sensor node. We then undertake an investigation to determine whether we can successfully identify into which electrical device the node is inserted and recognize electrical events related to the device by analyzing the electrical current data obtained from the sensor node. Although many portable electrical devices are now driven by internal rechargeable batteries, the market for primary cells is large and still growing. (about $16 billion in 2010) Also, rechargeable AA and AAA batteries (secondary cells) are widely used in our daily lives. Now, we are living surrounded by many electrical devices that are driven by D, C, AA, and AAA batteries. Moreover, due to the unreliable power supply caused by earthquake related accidents at nuclear power plants in Japan, battery-powered devices have been attracting attention. Also, several countries are reconsidering or have decided to decommission nuclear power plants, and so the value of batteries, which are very stable power sources, will increase. In addition, we can add extra value to conventional batteries by incorporating sensors.

4.1 Design

As mentioned in section 3, a battery-shaped sensor node designed based on the mimic sensor framework will be equipped with an electrical discharge function. That is, the sensor node will include a battery. Also, the sensor node measures an electrical current passing through the node just like an ammeter. In addition, the sensor node measures the voltage of the battery included in the node. We explain later why we also measure the voltage. Fig. 1 (a) is a schematic of our prototype battery-shaped sensor node. The sensor node includes a battery and a resistance. The node measures a current passing through the resistance. The node also measures the voltage of the included battery. The sensor node samples

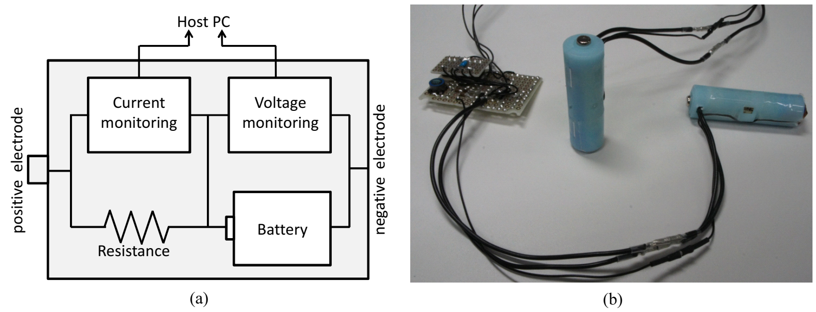
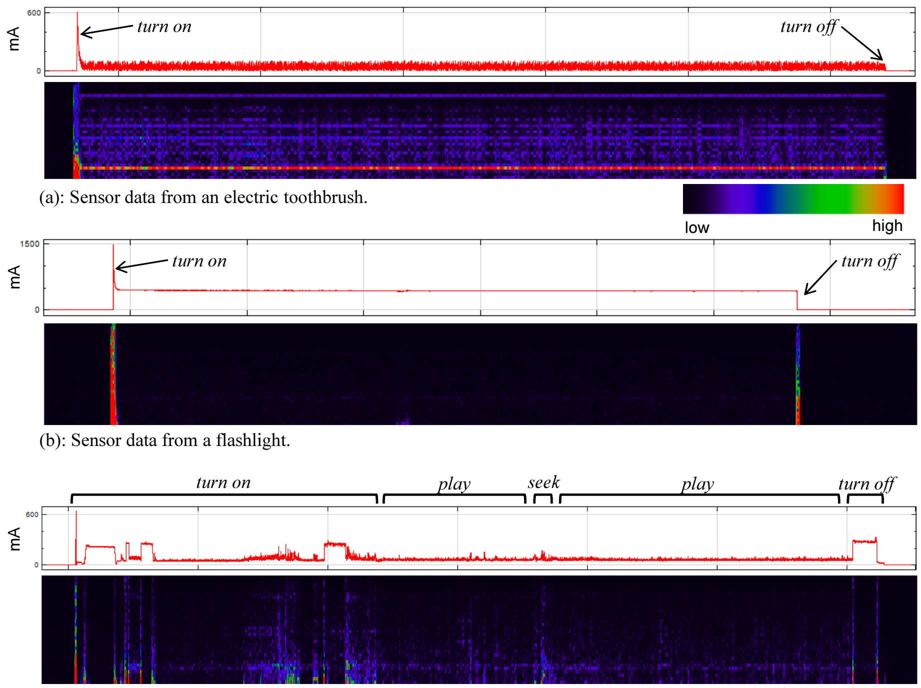


Fig. 1. (a) Schematic of our prototype sensor node. (b) Our prototype battery-shaped sensor nodes and a sensor board.

the current and voltage data at about 1000 Hz. Because the sensor node is a prototype, the node sends the sensor data to a host PC via cables and a sensor board. Fig. 1 (b) shows prototype AA battery-shaped sensor nodes and the sensor board. Because the node is a prototype, we simply incorporate a AAA or N battery in the node. We have also developed D and C battery-shaped sensor nodes. We use the prototype node to measure actual sensor data from various electrical devices and to investigate our device identification and event recognition method introduced in section 5.

4.2 Sensor data

The prototype sensor node measures an electrical current passing through the node. Fig. 2 shows several sets of example time-series sensor data obtained when we insert the node into various electrical devices and then operate the devices. The upper graph in Fig. 2 (a) shows time-series sensor data obtained from an electric toothbrush. The x-axis indicates time and the y-axis indicates the current sensor data value (mA). Just after the toothbrush was turned on, we observed an inrush of current. Then, we can find the characteristic frequency while the toothbrush was running, which was caused by the motor incorporated in the toothbrush. The lower graph in Fig. 2 (a) shows a frequency spectrogram computed from the time-series sensor data. We can see narrow peaks, which can be discriminative features, caused by the characteristic frequency. The upper graph in Fig. 2 (b) shows time-series sensor data obtained from a flashlight. When the flashlight was turned on, we also observed an inrush of current. Then, the sensor data values become static. The lower graph in Fig. 2 (b) shows a frequency spectrogram. While the flashlight was lit, there was no peak in the spectrogram because the sensor data values were static. We consider that the sensor data values obtained when the flashlight is lit can be simply used as a characteristic attribute of the flashlight. The upper graph in Fig. 2 (c) shows time-series sensor data obtained from a CD player. The graph also shows electrical events such as 'turn on,' 'play,' 'seek,' and 'turn off' related to the CD player. Unlike the toothbrush and flashlight, it took several seconds for the CD player to start operating. Then, the CD player started to play music. When a user selected the FF or RW



(a): Sensor data from an electric toothbrush.

(b): Sensor data from a flashlight.

(c): Sensor data from a CD player.

Fig. 2. Example current sensor data. The upper graphs show original time-series current data. The lower graphs show frequency spectrograms.

button, we observed a noise corresponding to seeking the next/previous track. It also took several seconds for the player to stop operating. The lower graph in Fig. 2 (c) shows a frequency spectrogram. As shown in these graphs, the sensor data from the CD player were time varying, and so we should model temporal changes in the sensor data to recognize electrical events related to the CD player.

Some portable electrical devices such as electric toothbrushes, electric shavers, handheld cleaners, and electric screwdrivers include motors. The rotation of the motors is impeded by various objects and phenomena. For example, motors in certain electrical devices are affected by the gravity of the earth, and so the way that the motor rotation changes may depend slightly on their posture in relation to the direction of the gravitational force. That is, the current sensor data of several electrical devices with motors will change depending on device posture. The upper graph in Fig. 3 shows time-series sensor data obtained from an electric toothbrush. Also, the lower graph in Fig. 3 shows the corresponding frequency spectrogram. We rotated the toothbrush 90 degrees while it was running to change its posture. In the spectrogram, we find that the peak frequency changed slightly when we rotated the toothbrush. This was caused by the gravity of the earth (and the posture change of the toothbrush). That is, when we

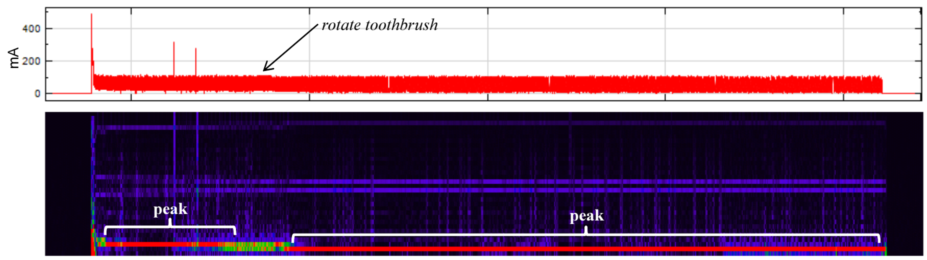


Fig. 3. Sensor data obtained from a toothbrush. We changed the posture of the toothbrush once while the toothbrush was running.

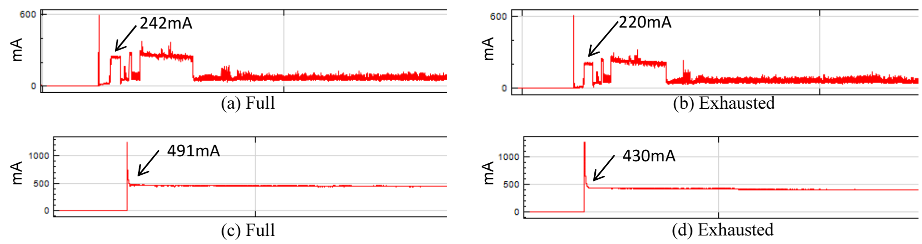


Fig. 4. Sensor data obtained from a CD player (a, b) and a flashlight (c, d) when we used fully charged batteries or exhausted batteries.

recognize electrical events related to motors with machine learning approaches, we should collect training sensor data of such devices under various conditions, e.g., by changing the postures of the devices as in actual use.

Here, we focus on battery-powered electrical devices. With use, the voltage of the battery decreases and this affects the electrical current passing through the device (and our sensor node). Fig. 4 shows current sensor data obtained from a CD player and a flashlight, both of which employ two AA batteries. The graph in Fig. 4 (a) shows sensor data obtained from the CD player when we used two fully charged batteries with a total voltage of 3.097 V. By contrast, the graph in Fig. 4 (b) shows sensor data obtained from the CD player when we used two exhausted batteries with a total voltage of 2.689 V. Although the two time-series sequences are similar, the sensor data values in Fig. 4 (b) are slightly smaller than those in Fig. 4 (a) even though we used the same CD player. For example, the current values just after the player was switched on, as indicated by the arrows in Fig. 4 (a) and (b), were about 242 and 220 mA, respectively, and they were slightly different. Also, the graph in Fig. 4 (c) shows sensor data obtained from the flashlight when we used two fully charged batteries with a total voltage of 3.161 V. On the other hand, the graph in Fig. 4 (d) shows sensor data obtained from the flashlight when we used two exhausted batteries whose total voltage was 2.654 V. The current values while the flashlight was lit in Fig. 4 (c) and (d) were about 491 and 430 mA, respectively. As above, the current

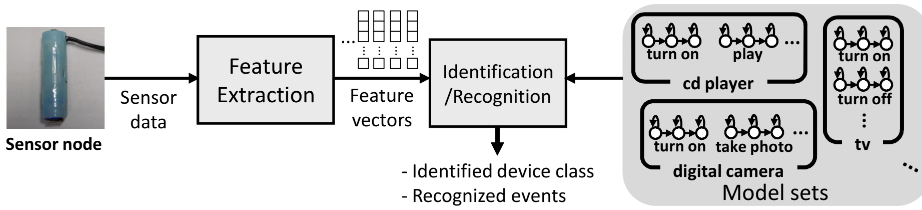


Fig. 5. Architecture of our identification/recognition method.

sensor data values change depending on the voltage values of the batteries. We need to cope with this problem when we identify electrical devices by employing machine learning approaches. There are two solutions to the problem. The first is to collect training sensor data from an electrical device by using batteries with various voltage values. The second is to compute a feature specific to the device and independent of battery voltage.

5 Our Method

By analyzing sensor data obtained from our sensor node, we try to identify in which electrical device the sensor node is inserted and recognize electrical events related to the device. Fig. 5 shows the architecture of our identification/recognition method. We first extract features from sensor data obtained from a sensor node. Then, we identify the electrical device and recognize electrical events by employing the extracted feature vectors. To achieve device identification and event recognition, we compare the vectors with a model set prepared for each type (product model) of electrical device constructed by employing a hidden Markov model (HMM). We describe our method in detail below.

5.1 Feature extraction

We assume time-series current data, and so we compute a feature vector for each sliding time window. We extract features based on the FFT components of 64 sample time windows. As mentioned in section 4.2, the FFT components and simple current values can be distinguishable features. Therefore, we use the computed FFT component values and the mean in each window as features. In addition to these features, we use the variance and energy, which can capture the intensity of sensor data changes, computed in each window as features. The energy feature is calculated by summing the magnitudes of squared discrete FFT components. For normalization, the sum was divided by the window length. Note that the DC component of the FFT is excluded from this summation.

In section 4.2, we mentioned that the current sensor data change depending on the battery voltage. To cope with this problem, we simply take account of the electrical resistance of the electrical device. The relationship between the voltage of a battery V, the current passing through the battery I, and the resistance of the device R when a certain electrical event occurs is described as $V = IR$. Because $R = V/I$ depends only on the device (electrical event), we use R as a feature. Our sensor node can measure both current and voltage, and so we can compute R. Note that because R reaches an infinite value when I is zero, we actually use $1/R$ as a feature. We describe how well this approach works by using the examples shown in Fig. 4. We first focus on the CD player sensor data in Fig. 4 (a) and (b). The current values indicated by the arrows in Fig. 4 (a) and (b) are about 242 and 220 mA, respectively (10.0% difference). On the other hand, the computed $1/R$ values in Fig. 4 (a) and (b) are 0.0782 and 0.0819, respectively (4.6% difference). With the flashlight, the current values obtained while the flashlight was lit shown in Fig. 4 (c) and (d) were about 491 and 430 mA, respectively (14.2% difference). On the other hand, the computed $1/R$ values in Fig. 4 (c) and (d) are 0.155 and 0.160, respectively (3.2% difference). It may be very difficult to reduce the error (difference) to zero because of the effects of many phenomena such as the voltage difference between the battery in our node and other batteries in the same device, the conversion characteristics of the regulator IC in electrical devices, and the ambient temperature. However, we consider that, simply by employing $1/R$, we can reduce the effect of the different battery voltages. Note that with an electrical device that can include n batteries connected in series, the value of I/V computed from the voltage and current values sensed by a sensor node in the device actually corresponds to n/R. This is because our sensor node can measure the voltage value of just the battery included in the node. As above, we extract a total of 36 features (32+1+1+1+1) from each window.

5.2 Hybrid identification/recognition method with HMMs

A feature vector sequence is extracted from sensor data obtained from a sensor node. The task is to identify in which electrical device the node is inserted and recognize electrical events related to the device. As shown in Fig. 5, our method identifies the device and simultaneously recognizes electrical events by comparing the sequence with model sets of electrical devices prepared in advance. We train the model sets by using labeled training data collected in advance. A model set is prepared for each type of electrical device and consists of left-to-right HMMs prepared for each electrical event. The HMMs allow us to capture the temporal regularity of events.

We explain our identification/recognition method in detail. We focus on a model set and recognize a feature vector sequence obtained from a node by using the model set. That is, we assume that the node is inserted into the electrical device related to the model set, and then recognize the sequence by using the HMMs in the model set. For the recognition, we simply use the Viterbi algorithm to find the most probable state sequence in/across the HMMs [18] and to compute the likelihood (score) of the state sequence. From the state sequence, we can know into which HMM (electrical event class) a feature vector at time t is classified. Here, because the model set (electrical device) includes multiple HMMs (electrical events), we assume state transitions across the HMMs. That is, we take account of a state transition from the last state of an arbitrary HMM to the first state of another HMM. With a CD player model set, for example, it corresponds to a state transition from a 'play' HMM to a 'turn off' HMM (and to all other HMMs in the model set). By taking the above state transition into account, we can represent transitions of electrical events. Here, we can specify state transitions among HMMs by using a handcrafted grammar. With a digital camera model set, for example, we can specify that an 'on' event (power ON state) must occur just after a 'turn on' event. We construct such a grammar for each model set (electrical device product) and investigate its effect in the next section.

We mentioned that the Viterbi algorithm outputs the most probable state sequence and its score when we recognize the feature vector sequence obtained from the node with a model set. So, we compute the most probable state sequence and its score when we recognize the feature vector sequence with each model set, and we decide that the node should be inserted into an electrical device (model set) corresponding to the highest score. As above, we can identify into which electrical device the node is inserted and recognize electrical events related to the device at the same time.

6 Evaluation

6.1 Data set

For the evaluation, we prepared the many portable electrical devices listed in Table 1. We selected these devices from those frequently found in appliance and online stores. Table 1 also shows electrical events related to each device. Each device includes an 'off' event that means the power OFF state. In addition, it takes several seconds for devices such as TVs, CD players, and digital cameras to start. Such devices include 'turn on' events. On the other hand, it takes a very short time for such devices as flashlights and toothbrushes to start. We do not consider that such devices include 'turn on' events because it is very difficult to annotate such short events. We also ignored events (functions) where the electrical current values remain unchanged. For example, the 'zoom' functions of the digital cameras and digital camcorder used in our experiment did not induce any change in the current values from those of 'on' events (power ON state), and so we regard 'zoom' events to be included in 'on' events. This may be because the regulator IC in the device could provide sufficient current required for the event, and so the current that the batteries supplied to the regulator did not change. In addition, with electrical devices with a large number of functions, we focus only on the main events (functions).

We obtained training data in our experimental environment by using the devices listed in Table 1. We inserted one sensor node into an electrical device to collect data. As mentioned in section 4.2, we collected training data by changing the postures of the devices as if in actual use. We also changed the battery voltages of the node. We used each device about 30 times in total. We collected test data in real three houses (houses A, B, and C). In section 4.2 we showed that sensor data are affected by changes in the posture of electrical devices,

Table 1. Electrical devices used in our experiment and their electrical events.

device	events	device	events
digital camera 1	on, turn on, turn off, take photo, focus, off	cd player 1	turn on, turn off, play, seek, off
digital camera 2	on, turn on, turn off, take photo, focus, off	cd player 2	turn on, turn off, play, seek, off
digital camcorder	on, turn on, turn off, take video, off	tv 1	turn on, turn off, show, off
vacuum 1	vacuum, off	tv 2	turn on, show, off
vacuum 2	vacuum, off	lantern	light, off
video game	on, off	flashlight 1	light, off
shaver 1	shave, off	flashlight 2	light, off
shaver 2	shave, off	cassette player 1	ff/rw, play, off
shaver 3	shave, off	cassette player 2	ff/rw, play, off
screwdriver 1	ff/rw, light, off	dvd player	turn on, play, off
screwdriver 2	ff/rw, off	soldering iron	on, off
toothbrush 1	brush, off	mill	coarse, fine, off
toothbrush 2	brush, off	toy 1 (ship)	go ahead, off
toothbrush 3	brush, off	toy 2 (car)	go forwards, go backwards, off

and so we tested the use of devices by different participants. We gave several different electrical devices to a participant in each house (House A: 10 devices, house B: 10 devices, house C: 8 devices. See Tables 3, 4, and 5.) and asked the participant to use the devices equipped with our nodes. We collect sensor data by using a semi-naturalistic collection protocol [1] that permits greater variability in participant behavior than laboratory data. In the protocol, each participant took part in a session that included the use of electrical devices in a random sequence following instructions on a worksheet. The participants were relatively free as regards how they used each device because the instructions on the worksheet were not very strict, e.g., "play music freely with a CD player" and 'watch an arbitrary TV channel(s).' Because our prototype devices were connected to a host PC via cables, they were used in the same room in each house. However, we asked the participants to employ the devices as in actual use. Note that, as regards the soldering iron, the participant simply turned it on and did not solder anything. Each participant completed a total of ten sessions. (A and B: 10 devices$\times$10 sessions, C: 8 devices$\times$10 sessions) Each session lasted about ten minutes. We used fully charged batteries in the first house, and then used the same partly used batteries in the next house. Note that, when a battery became very weak, we replaced it with new one. We observed the participants by using video cameras to enable us to annotate the obtained data.

6.2 Evaluation methodology

We constructed a model set by using training data obtained in our experimental environment, and evaluated the performance of our method by using test data obtained in the participants' houses. That is, we assumed that end users did not prepare training data in their houses. To investigate the effectiveness of our approach, we tested the following eight methods.

- HMM: This method models electrical events with HMMs as mentioned in section 5.2. This method does not use $1/R$ as a feature. Also, this method assumes

that the transition probabilities across HMMs are all identical.
- HMM(grammar): This method models electrical events with HMMs. This method does not use $1/R$ as a feature. Also, this method employs a handcrafted grammar prepared for each device when computing transitions among HMMs. We provide some grammar examples written in extended BNF. The grammar for flashlights is described as ('off' {'light' 'off'}). With this grammar 'off' and 'light' events are defined as being alternately repeated. The grammar for a digital camera is described as ({'off' 'turn on' 'on' {'take photo'|'focus'|'on'} 'turn off' } 'off'). With this grammar the definition is that 'off,' 'turn on,' and 'on' events occur sequentially and then 'take photo,' 'focus,' and 'on' events are repeated alternately before the 'turn off' and 'off' events occur. More specifically, for example, the transition probabilities from the last state of the 'turn on' HMM to the first states of other HMMs except for the 'on' HMM are defined as zero. By contrast, the transition probability to the 'on' HMM is one. Also, the transition probabilities from the last state of the 'take photo' HMM to 'focus,' 'on,' and 'turn off' are identical (1/3). By contrast, the transition probabilities to the other HMMs are zero.
- SVM: This method uses the SVM in place of the HMM. We construct a classifier for device identification that classifies each feature vector into an electrical device class. We classify feature vectors obtained from a sensor node and then we identify the device class of the node by using the majority voting of the feature vector classification results. Note that when we construct the classifier, we ignore feature vectors whose current mean values are zero, which corresponds to 'off' events, because 'off' events are included in all device classes. We also construct a classifier for event recognition for each electrical device product. The classifier classifies each feature vector into an electrical event class. Unlike HMM-based methods, this method cannot model the temporal regularity of electrical events. This method also does not use $1/R$ as a feature.
- Tree: This method uses a decision tree in place of the SVM in the above SVM method. This method does not use $1/R$ as a feature.
- HMM-R: This method uses $1/R$ as a feature. Also, this method assumes that the transition probabilities across HMMs are all identical.
- HMM-R(grammar): This method uses $1/R$ as a feature. This method also employs handcrafted grammar.
- SVM-R: This method uses $1/R$ as a feature. This method also uses the SVM in place of the HMM.
- Tree-R: This method uses $1/R$ as a feature. This method also uses the decision tree in place of the HMM.

6.3 Results

Device identification Fig. 6 shows the transitions of device identification accuracies when we increase the number of test sessions used to identify devices. When the number of sessions (#sessions) is three, for example, we identify devices by using only the sensor data obtained in each house in the first three sessions. Because we prepared model sets of 28 electrical devices, the random guess

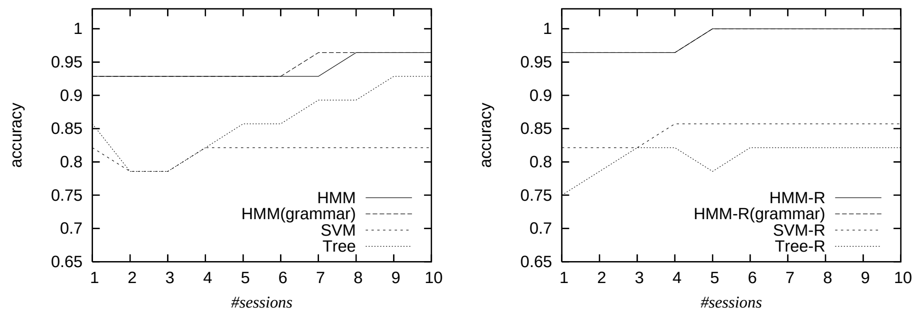


Fig. 6. Transitions of device identification accuracies when we increase #sessions that are used to identify devices.

Table 2. Event recognition accuracies (average F-measure) in each house.

	House A	House B	House C	AVG.
HMM	0.795	0.807	0.839	0.814
HMM(grammar)	0.861	0.822	0.853	0.845
SVM	0.823	0.834	0.837	0.831
Tree	0.834	0.773	**0.908**	0.839
HMM-R	0.816	0.809	0.843	0.823
HMM-R(grammar)	**0.872**	**0.865**	0.875	**0.871**
SVM-R	0.821	0.828	0.839	0.829
Tree-R	0.822	0.779	0.889	0.830

ratio is only 3.6% ($1/28 = 0.036$). However, the HMM and HMM(grammar) methods achieved 96.4% accuracies when #sessions was ten. These methods greatly outperformed SVM and Tree, which cannot capture the temporal regularity of electrical events. The HMM-R and HMM-R(grammar) methods achieved 100% accuracies when #sessions was larger than four. (The transitions of these methods were completely identical in the right graph in Fig. 6.) By using the $1/R$ feature, we could identify devices perfectly. These methods also greatly outperformed SVM-R and Tree-R. That is, we confirmed the importance of capturing the temporal regularity of electrical events and taking the electrical resistances of electrical devices into account. Basically, a larger #sessions exhibited greater identification accuracy because we could use sufficient quantities of sensor data and capture the discriminative features of the sensor data. However, the HMM-R and HMM-R(grammar) methods achieved 96.4% accuracies even when #sessions was one. These results indicate that even if a battery sensor node is removed from an electrical device and then inserted into another device, these methods can soon identify the new device.

Event recognition To evaluate the event recognition performance, we assumed that the device identification results were all correct, and then calculated the precision, recall, and F-measure based on the results for the estimated event class

Table 3. Event recognition accuracies in house A.

		HMM			HMM-R			HMM-R(grammar)		
		precision	recall	F-measure	precision	recall	F-measure	precision	recall	F-measure
digital camera 2	on	0.874	0.671	0.759	0.895	0.716	0.796	0.887	0.726	0.799
	turn on	0.473	0.561	0.513	0.493	0.593	0.538	0.978	0.714	0.826
	turn off	0.316	0.391	0.349	0.335	0.396	0.363	1	0.941	0.97
	take photo	0.356	0.408	0.38	0.377	0.415	0.395	0.518	0.776	0.622
	focus	0.362	0.694	0.476	0.378	0.694	0.489	0.44	0.898	0.591
	off	0.975	0.975	0.975	0.975	0.975	0.975	0.975	0.975	0.975
vacuum 2	vacuum	0.993	0.996	0.994	0.993	0.996	0.994	0.993	0.996	0.994
	off	0.988	0.977	0.982	0.988	0.977	0.982	0.988	0.977	0.982
video game	on	1	0.939	0.968	1	0.983	0.992	1	0.983	0.992
	off	0.83	1	0.907	0.947	1	0.973	0.947	1	0.973
shaver 1	shaving	1	0.987	0.994	1	0.986	0.993	1	0.986	0.993
	off	0.925	1	0.961	0.919	1	0.958	0.919	1	0.958
screw-driver 1	ff/rw	0.997	0.982	0.99	1	0.982	0.991	1	0.982	0.991
	light	0.988	0.995	0.992	0.987	0.995	0.991	0.987	0.995	0.991
	off	0.976	0.979	0.977	0.976	0.979	0.977	0.976	0.979	0.977
tooth-brush 1	brush	1	0.994	0.997	1	0.994	0.997	1	0.994	0.997
	off	0.972	1	0.986	0.976	1	0.988	0.976	1	0.988
cd player 1	turn on	0.484	0.51	0.497	0.515	0.579	0.545	1	0.569	0.726
	turn off	0.287	0.566	0.381	0.482	0.762	0.59	0.621	0.979	0.76
	play	0.961	0.65	0.775	0.956	0.686	0.799	0.946	0.898	0.922
	seek	0.047	0.66	0.088	0.06	0.681	0.11	0.075	0.936	0.139
	off	0.995	0.996	0.995	0.995	0.999	0.997	0.991	1	0.995
tv 1	turn on	0.73	0.514	0.603	0.731	0.651	0.689	0.75	0.982	0.851
	turn off	0.309	0.488	0.379	0.336	0.382	0.357	0.917	0.089	0.163
	show	1	0.987	0.994	1	0.994	0.997	0.999	0.999	0.999
	off	0.795	1	0.886	0.866	1	0.928	0.918	1	0.957
lantern	light	0.999	0.994	0.997	0.999	0.994	0.997	0.999	0.994	0.997
	off	0.97	0.997	0.983	0.97	0.997	0.983	0.97	0.997	0.983
cassette player 2	ff/rw	0.936	0.945	0.94	0.938	0.985	0.961	0.938	0.985	0.961
	play	0.962	0.947	0.954	0.991	0.947	0.968	0.991	0.947	0.968
	off	0.976	1	0.988	0.976	1	0.988	0.976	1	0.988
AVG.		0.790	0.832	0.795	0.808	0.850	0.816	0.893	0.913	0.872

at each time slice. Also, the precision, recall, and F-measure were computed by using all test data (all ten sessions) obtained in each house. Table 2 shows the event recognition accuracies of the three houses for each recognition method. The HMM results were poorer than those of SVM and Tree, which are the discriminative models. This may be because the classification performance of the discriminative techniques, which find the discriminant boundaries of the classes, is often superior to that of generative models such as the HMM. On the other hand, HMM(grammar) outperformed HMM, SVM, and Tree. As described in detail later, the method was good at recognizing events of electrical devices that produced confusing sensor data patterns by using the handcrafted grammar. HMM-R(grammar), which uses both the grammar and the $1/R$ feature, achieved the highest accuracy.

Tables 3, 4, and 5 show the detailed event recognition accuracies obtained in each house. Because most electrical devices have only two electrical events, the event recognition accuracies were high in each house. We first focused on the HMM and HMM-R methods. By using the $1/R$ feature, we could slightly improve the accuracies of almost all the events. We then focused on the HMM-R and HMM-R(grammar) methods. By using the grammar, we could greatly improve the accuracies of the highly functional devices, namely CD players, TVs, DVD players, digital cameras, and a digital camcorder, which have many electrical event classes. The average improvement as regards F-measure was 0.122. As shown in Fig. 2 (c), 'turn on' and 'turn off' events of these devices have complex time-series sensor data. Therefore, the recognition accuracies with the HMM-R method related to these events were poor. The average as regards F-measure was

Table 4. Event recognition accuracies in house B.

		HMM			HMM-R			HMM-R(grammar)		
		precision	recall	F-measure	precision	recall	F-measure	precision	recall	F-measure
digital camcorder	on	0.902	0.994	0.946	0.886	0.976	0.929	0.884	0.976	0.928
	turn on	0.663	0.757	0.707	0.676	0.77	0.72	1	0.747	0.855
	turn off	0.278	0.252	0.264	0.282	0.256	0.268	0.659	0.966	0.784
	take video	0.965	0.649	0.776	0.906	0.611	0.73	0.907	0.703	0.792
	off	0.979	0.998	0.988	0.979	1	0.989	0.976	1	0.988
shaver 2	shave	1	0.994	0.997	0.997	0.994	0.995	0.997	0.994	0.995
	off	0.984	1	0.992	0.984	0.992	0.988	0.984	0.992	0.988
screw-driver 2	ff/rw	1	0.982	0.991	1	0.982	0.991	1	0.982	0.991
	off	0.972	1	0.986	0.972	1	0.986	0.972	1	0.986
tooth-brush 2	brush	1	0.992	0.996	1	0.992	0.996	1	0.992	0.996
	off	0.983	1	0.991	0.983	1	0.991	0.983	1	0.991
cd player 2	turn on	0.838	0.392	0.534	0.93	0.422	0.581	0.987	0.487	0.652
	turn off	0.387	0.765	0.514	0.506	0.824	0.627	0.674	0.948	0.788
	play	0.924	0.945	0.935	0.916	0.967	0.941	0.915	0.969	0.941
	seek	0.018	1	0.035	0.019	1	0.037	0.02	1	0.04
	off	0.995	0.982	0.989	0.996	0.995	0.996	0.986	0.995	0.99
flash-light 1	light	1	0.99	0.995	1	0.99	0.995	1	0.99	0.995
	off	0.975	1	0.987	0.975	1	0.987	0.975	1	0.987
flash-light 2	light	1	0.986	0.993	1	0.987	0.994	1	0.987	0.994
	off	0.973	1	0.986	0.976	1	0.988	0.976	1	0.988
cassette player 1	ff/rw	0.94	0.239	0.381	0.543	0.038	0.071	0.543	0.038	0.071
	play	0.551	0.908	0.686	0.492	0.912	0.639	0.492	0.912	0.639
	off	0.798	0.976	0.878	0.822	0.998	0.901	0.822	0.998	0.901
dvd player	turn on	0.7	0.742	0.721	0.788	0.785	0.787	0.998	0.992	0.995
	play	0.392	0.343	0.366	0.56	0.564	0.562	0.984	0.996	0.99
	off	0.989	0.989	0.989	0.989	0.989	0.989	0.989	0.989	0.989
soldering iron	on	0.994	0.992	0.993	0.994	0.992	0.993	0.994	0.992	0.993
	off	0.982	0.986	0.984	0.982	0.986	0.984	0.982	0.986	0.984
AVG.		0.828	0.852	0.807	0.827	0.858	0.809	0.882	0.915	0.865

Table 5. Event recognition accuracies in house C.

		HMM			HMM-R			HMM-R(grammar)		
		precision	recall	F-measure	precision	recall	F-measure	precision	recall	F-measure
digital camera 1	on	0.63	0.929	0.751	0.636	0.956	0.764	0.629	0.956	0.759
	turn on	0.556	0.599	0.577	0.675	0.574	0.621	1	0.669	0.801
	turn off	0.611	0.374	0.464	0.702	0.586	0.639	0.995	0.747	0.854
	take photo	0.658	0.554	0.602	0.667	0.577	0.618	0.803	0.869	0.835
	focus	0.397	0.818	0.535	0.382	0.788	0.515	0.311	0.848	0.455
	off	0.984	0.992	0.988	0.968	0.992	0.98	0.976	1	0.988
vacuum 1	vacuum	0.998	0.993	0.996	0.998	0.993	0.996	0.998	0.993	0.996
	off	0.981	0.995	0.988	0.981	0.995	0.988	0.981	0.995	0.988
shaver 3	shave	1	0.994	0.997	1	0.994	0.997	1	0.994	0.997
	off	0.987	1	0.993	0.987	1	0.993	0.987	1	0.993
tooth-brush 3	brush	0.998	0.989	0.994	0.998	0.989	0.994	0.998	0.989	0.994
	off	0.974	0.996	0.985	0.974	0.996	0.985	0.974	0.996	0.985
tv 2	turn on	0.716	0.536	0.613	0.751	0.53	0.621	0.892	0.484	0.627
	show	0.886	0.92	0.903	0.881	0.95	0.914	0.875	0.976	0.922
	off	0.578	0.965	0.723	0.525	0.625	0.571	0.552	0.98	0.706
mill	coarse	1	0.955	0.977	1	0.963	0.981	1	0.963	0.981
	off	0.977	1	0.988	0.977	1	0.988	0.977	1	0.988
toy 1	go ahead	1	0.994	0.997	1	0.995	0.997	1	0.995	0.997
	off	0.971	1	0.985	0.977	1	0.988	0.977	1	0.988
toy 2	go forwards	0.684	0.849	0.757	0.67	0.858	0.752	0.67	0.858	0.752
	go backwards	0.801	0.563	0.661	0.812	0.563	0.665	0.812	0.563	0.665
	off	0.949	1	0.974	0.975	1	0.987	0.975	1	0.987
AVG.		0.833	0.864	0.839	0.843	0.860	0.843	0.881	0.903	0.875

0.571. We could greatly improve the accuracies by incorporating such grammar as 'turn on' event follows 'off' event and 'off' event follows 'turn off' event. The average improvement was 0.248.

Finally, we describe the electrical events that HMM-R(grammar) could not recognize with high accuracy. The accuracies as regards the 'seek' events for the two CD players in houses A and B were very poor. This may be because we could not prepare sufficient quantities of training data (feature vectors) about the events. As shown in Fig. 2 (c), the time length of 'seek' is very short. The accuracy as regards the 'turn off' event of tv 1 in house A was poor. The event was mistakenly confused with an 'off' event because the current value approaches zero during a 'turn off' event. Also, the accuracy as regards the 'ff/rw' event of cassette player 1 in house B was poor. Because the player was light in weight, we collected training data by holding it in the hand. However, the participant placed the player on a table and operated it. (We did not collect training data when the player was placed on a table.) As mentioned in section 4.2, the motor rotation is affected by the posture of the motor. The HMMs of the player trained with our training data could not capture the use of the player in house B.

7 Conclusion

In this paper, we proposed and implemented a prototype battery-shaped sensor node for monitoring the use of electrical devices. We also proposed a device identification and electrical event recognition method by analyzing the sensor data. With the method, we can automatically identify into which electrical device the sensor node has been inserted and recognize electrical events related to the device. In addition, we achieved very high identification and recognition accuracies by using sensor data obtained from three real houses. As a part of our future work, we plan to develop other types of sensor nodes based on the mimic sensor framework to capture a broader range of real world activities. We also plan to develop wireless battery-shaped sensor nodes and conduct long-term experiments using them. We consider that there are two problems as regards developping wireless sensor nodes. The first problem relates to the size of the sensor and wireless transmission components. However, our sensing architecture is very simple as shown in Fig. 1 (a). Also, recently, a CPU has become available that includes a wireless transmission component the length of whose side is less than the diameter of an AA battery. Moreover, an SD card that includes a CPU and a WiFi AP component is already on the market (e.g., Eye-Fi[3]). The second problem relates to the energy consumption of the node. We can greatly reduce the energy consumption by stopping the node from transmitting sensor data while the current sensor data value is zero, i.e., OFF state. Also, as shown in Figs. 2 and 3, many electrical devices continually produce similar sensor data patterns. The node should be designed to transmit sensor data only when they are very different from those of the latest sample.

References

1. L. Bao and S.S. Intille. Activity recognition from user-annotated acceleration data. In *Pervasive 2004*, pages 1–17, 2004.
2. C. Beckmann, S. Consolvo, and A. LaMarca. Some assembly required: Supporting end-user sensor installation in domestic ubiquitous computing environments. In *UbiComp 2004*, pages 107–124, 2004.
3. G. Cohn, S. Gupta, J. Froehlich, E. Larson, and S. Patel. GasSense: Appliance-level, single-point sensing of gas activity in the home. In *Pervasive 2010*, pages 265–282, 2010.

[3] http://eye.fi/

4. J.E. Froehlich, E. Larson, T. Campbell, C. Haggerty, J. Fogarty, and S.N. Patel. Hydrosense: Infrastructure-mediated single-point sensing of whole-home water activity. In *Ubicomp 2009*, pages 235–244, 2009.
5. S. Gupta, M.S. Reynolds, and S.N. Patel. ElectriSense: Single-point sensing using EMI for electrical event detection and classification in the home. In *Ubicomp 2010*, pages 139–148, 2010.
6. X. Jiang, S. Dawson-Haggerty, P. Dutta, and D. Culler. Design and implementation of a high-fidelity ac metering network. In *IPSN 2009*, pages 253–264, 2009.
7. Y. Kim, T. Schmid, Z.M. Charbiwala, and M.B. Srivastava. ViridiScope: design and implementation of a fine grained power monitoring system for homes. In *Ubicomp 2009*, pages 245–254, 2009.
8. J. Lifton, M. Feldmeier, Y. Ono, C. Lewis, and J.A. Paradiso. A platform for ubiquitous sensor deployment in occupational and domestic environments. In *IPSN 2007*, pages 119–127, 2007.
9. B. Logan, J. Healey, M. Philipose, E.M. Tapia, and S. Intille. A long-term evaluation of sensing modalities for activity recognition. In *Ubicomp 2007*, pages 483–500, 2007.
10. P. Lukowicz, H. Junker, M. Stager, T. Von Buren, and G. Tröster. Wearnet: A distributed multi-sensor system for context aware wearables. In *Ubicomp 2002*, pages 361–370, 2002.
11. T. Maekawa, Y. Kishino, Y. Sakurai, and T. Suyama. Recognizing the use of portable electrical devices with hand-worn magnetic sensors. In *Pervasive 2011*, pages 276–293, 2011.
12. T. Maekawa and S. Watanabe. Unsupervised activity recognition with user's physical characteristics data. In *Int'l Symp. on Wearable Computers*, pages 89–96, 2011.
13. T. Maekawa, Y. Yanagisawa, Y. Kishino, K. Ishiguro, K. Kamei, Y. Sakurai, and T. Okadome. Object-based activity recognition with heterogeneous sensors on wrist. In *Pervasive 2010*, pages 246–264, 2010.
14. T. Maekawa, Y. Yanagisawa, Y. Kishino, K. Kamei, Y. Sakurai, and T. Okadome. Object-blog system for environment-generated content. *IEEE Pervasive Computing*, 7(4):20–27, 2008.
15. E.D. Mynatt, J. Rowan, S. Craighill, and A. Jacobs. Digital family portraits: Supporting peace of mind for extended family members. In *CHI 2001*, pages 333–340, 2001.
16. S.N. Patel, T. Robertson, J.A. Kientz, M.S. Reynolds, and G.D. Abowd. At the flick of a switch: Detecting and classifying unique electrical events on the residential power line. In *Ubicomp 2007*, pages 271–288, 2007.
17. M. Philipose, K.P. Fishkin, and M. Perkowitz. Inferring activities from interactions with objects. *IEEE Pervasive Computing*, 3(4):50–57, 2004.
18. L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.
19. N. Ravi, N. Dandekar, P. Mysore, and M.L. Littman. Activity recognition from accelerometer data. In *IAAI 2005*, volume 20, pages 1541–1546, 2005.
20. E.M. Tapia, S.S. Intille, and K. Larson. Activity recognition in the home using simple and ubiquitous sensors. In *Pervasive 2004*, pages 158–175, 2004.
21. E.M. Tapia, S.S. Intille, and K. Larson. Portable wireless sensors for object usage sensing in the home: challenges and practicalities. In *European Conference on Ambient Intelligence 2007*, pages 19–37, 2007.
22. T. van Kasteren, A. Noulas, G. Englebienne, and B. Kröse. Accurate activity recognition in a home setting. In *Ubicomp 2008*, pages 1–9, 2008.