

Physical activity recognition using a single triaxial accelerometer and a barometric sensor for baby and child care in a home environment

Yunyoung Nam^{a,*} and Jung Wook Park^b

^a *Department of Biomedical Engineering, Worcester Polytechnic Institute, MA, USA*
E-mail: ynam@wpi.edu

^b *U-Healthcare Department, BIT Computer Co., Ltd, Seoul, South Korea*
E-mail: ubihuman@gmail.com

Abstract. In this paper, we describe and evaluate an activity recognition system using a single 3-axis accelerometer and a barometric sensor worn on a waist of the body. The purpose of this work is to prevent child accidents such as unintentional injuries at home. In order to prevent child accidents in the home and reduce efforts of parents, we present a new safety management system for babies and children. We collected labeled accelerometer data from babies as they performed daily activities which are standing still, standing up, sitting down, walking, toddling, crawling, climbing up, climbing down, stopping, wiggling, and rolling. In order to recognize daily activities, mean, standard deviation, and slope of time-domain features are calculated over sliding windows. In addition, the FFT analysis is adopted to extract frequency-domain features of the aggregated data, and then energy and correlation of acceleration data are calculated. We used the resulting training data to induce a predictive model for activity recognition. Naive Bayes, Bayes Net, Support Vector Machine, k-Nearest Neighbor, Decision Tree, Decision Table, Multilayer Perceptron, Logistic classifiers are tested on these features. Classification results using training and eight classifiers were compared. The overall accuracy of activity recognition was 96.2% using only a single wearable triaxial accelerometer sensor with the k-Nearest Neighbor.

Keywords: Activity recognition, accelerometer, wearable device, baby care, child care

1. Introduction

Early work in ubiquitous computing dealt with anywhere and anytime communication with smart objects [1]. However its focus has been shifted to concerns over topics such as how it will change our lives [2]. Numerous studies have attempted to evaluate how computing works in various fields such as sensor-based abnormal activity detection for health care [3], the mobile sensing platform for diabetics, and a personal-fuel gauge [4]. Most of the previous stud-

ies have concentrated on human activity recognition as an essential technology for wellness applications. In this paper, we propose a human activity recognition method for accident prevention in the home environment. Since the activities of child are quite unpredictable, a parent should keep an eye on their child 24 hours a day. However, this may not always be practical. With this motivation, an assistive device would be a sensible solution to prevent home accidents without demanding absolute attention from parents.

Although numerous researchers have proposed various activity recognition methods, studies on human activity recognition, pervasive safety management sys-

* Corresponding author.

tem and multi-sensor fusion are challenging issues in terms of accurate recognition. In general, a pervasive safety management system aims to reduce risk factors of injuries to prevent accidents by using smart devices such as cameras and multiple sensors. In particular, some researchers attempted to assist child safety care based on activity recognition. Na et al. [5,6] proposed a smart vision sensor for detecting risk factors of a falling accident in a home environment. They used image-processing methods, namely clutter detection and toddler tracking, to recognize risk factors. Nishida et al. [7] conducted infant behavior simulation for safety management. This computational approach is taken in order to comprehensively understand the behavior of infants and children. This approach could also be utilized for the clarification of the dynamics of a system that includes behavior-related accidents.

The pursuit of low power distributed sensing under the user's natural physiological conditions has imposed significant challenges on integrating information from what is often heterogeneous, incomplete, and error-prone sensor data. Multi-sensor fusion, therefore, is essential to maximize information content and reduce both systematic and random errors when a lot of sensors are used for human activity recognition. Zouba et al. [8] proposed multi-sensor fusion using video cameras and environmental sensors for monitoring elderly activities. They choose to perform fusion at the event-level twelve household activities including: using the fridge, using the cupboards, using the microwave, preparing a mail, and so on. Zhu et al. [9] also suggested human activity recognition by fusing two wearable inertial sensors attached to one foot and the waist of a human subject, respectively.

In general, the nature of information interaction involved in sensor fusion can be classified as competitive, complementary, and cooperative fusion [10–12]. In competitive fusion, each sensor provides equivalent information about the process being monitored. It typically involves the handling of redundant, but sometimes inconsistent, measurement. The nature of competitive sensing means that it is ideally suited for multi-sensor calibration, consistency maximization, and fault tolerant sensing. In complementary fusion, sensors do not depend on each other directly, as each sensor captures different aspects of the physical process. The measured information is merged to form a more complete picture of the phenomenon. For example, multiple accelerometers could be attached to body parts for recognizing bowling activity. Each sensor observes disjoint motions such as inclined upper body, motionless

hands and legs, and nodding head. More attached accelerometers lead to a more complete recognition result.

In cooperative fusion, sensors work together to gather complex information that is difficult to obtain from the sensors individually. For example, in the case of a climbing up activity, an accelerometer could only recognize the activity as standing up and sitting down continuously. A barometric sensor could measure a variation of the height of the sensor from the ground, not the activity. Cooperative fusion of the two sensors enables recognition of the activity that could not be detected by each single sensor. Due to the compounding effect, the accuracy and reliability of cooperative fusion is sensitive to inaccuracies in all simple sensor components used. In this paper, we select the cooperative fusion model to combine information from sensors to capture data with improved reliability, precision, fault tolerance, and reasoning power to a degree that is beyond the capacity of each sensor.

In [13], eWatch devices were placed on the belt, shirt pocket, trouser pocket, backpack, and neck to recognize six activities. Each eWatch consisted of a biaxial accelerometer and a light sensor. In [4], they used a multimodal sensor device consisting of seven different types of sensors to recognize activities such as walking, sitting, standing, ascending stairs, descending stairs, riding an elevator up and down, and brushing one's teeth. Yang et al. [14] proposed a distributed sparsity classifier to recognize human activities using five wireless motion sensors. They represented and classified thirteen action classes using a set of 40-D locality preserving projection features accurately. Kasteren et al. [15] compared four probabilistic models such as hidden Markov model, hidden semi-Markov models, conditional random fields, and semi-Markov conditional random fields to validate the importance of duration modeling. They found that duration modeling could lead to significantly better performance.

In our previous work [16], we used a single tri-axial accelerometer, along with an embedded image sensor worn at the user's waist, to identify nine activities. In our previous work [17], we proposed an incremental statistical method to determine conflicts and to infer user intentions through analyzing the daily human activity patterns collected from physical sensors. Although these multi-sensor approaches do indicate the great potential of mobile sensor data as more types of sensors are being incorporated into devices, our approach shows that only one single unit of sensor nodes

will collect multiple types of information and recognize most daily activities for baby and child in a home environment.

This paper is organized as follows. Section 2 explains patterns of home accidents by statistical data and observation and presents an overview of the application model. Section 3 presents the outline of the proposed system, data collection, and data preprocessing. Section 4 describes the process for addressing the activity recognition task, including the activity recognition method. Section 5 describes the design of the proposed wearable device and shows our experimental results. Finally, we conclude the paper in Section 6.

2. Problem description and approach

2.1. Problem description and motivation

According to the statistical records [18,19] of US and UK, child home accidents occurred more frequently in the home environment. Based on survey in 2005, in South Korea, an average of 20.9% of children aged under 14 died as a result of falls [20]. Other leading accidents reported in this survey were poisoning, burns, and slipping. According to the child accident facts [21], which are surveyed by the CAPT in the UK, falls accounted for over 40% of all home accidents of children. It is unrealistic to ask parents keep their eyes on babies 24 hours a day, 7 days a week. In order to prevent child home accidents and reduce the efforts of parents, a new safety management method is required. Also early detection is needed to prevent unintentional injury at home. We have monitored behavioral characteristics when a child is in danger or in trouble by observing the child in a real home environment. As expected, the activities of a child were quite unpredictable. In the case of adults, their activities did not include unnecessary movement. If the child tries to reach somewhere, he moved with walking, toddling and sitting randomly, which makes child activity recognition more complicated.

From the observations in Fig. 1, one of most dangerous activities in our observation is climbing up on tables or windows ledges, since a child can be injured if he or she hits his or her head when falling from a chair or a table. As shown in Fig. 1c, the child used low chairs or large toys as a stool to climb up on a table. The mother was cleaning up a veranda while she was climbing up and she could not recognize the dangerous activity of her child. After playing, an activity

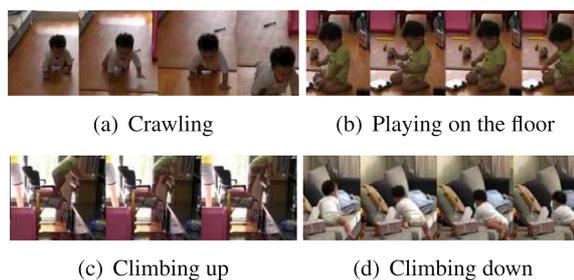


Fig. 1. Activities of a child in a living room.

of climbing down was observed in Fig. 1d. However, since no accurate tool has been developed for observing the wide variety of behaviors that occur in a living room, we lack quantitative data on this. This results in difficulty in comprehensively understanding baby activities. The behaviors addressed in this field are very limited when compared with the diversity of baby activities.

A baby performed various physical activities: wiggle, roll, crawl, climb, rock, bounce, rest, eat, make noise, grasp or mouth or drop things. As a baby learns to roll over, sit and crawl, his muscles will continue to strengthen. Between 8 and 10 months, he will probably start trying to pull himself up to stand while holding onto furniture. At this time, they are at risk of falling from furniture and raised surfaces, like changing tables and counter-tops. As older babies learn to crawl and walk, they are at risk of falling on the stairs. As toddlers learn to climb, they are at risk of falling out of windows and from furniture like bookcases. Since the major causes of fall-related injuries change as a child grows and develops, fall prevention needs to be addressed. These prevention strategies are not meant to take away physical activity opportunities, but rather to create a safe environment in which physical activity can take place.

2.2. Approach

In order to solve the problems as mentioned in the previous section, we have developed a wearable sensor device and a monitoring application to gather information and to recognize baby activities. We classified baby activities into eleven daily activities which are wiggling, rolling, standing still, standing up, sitting down, walking, toddling, crawling, climbing up, climbing down, and stopping. Vision-based activity recognition approaches [18,22] are not applicable in a home environment, since cameras present a privacy problem. The progress of an advanced sensor and SoC

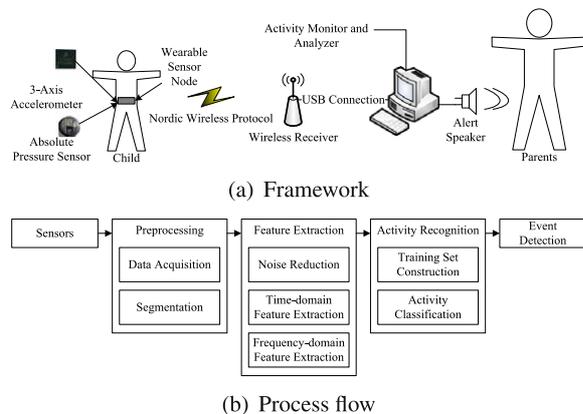


Fig. 2. Application environment of proposed framework and process flow.

technologies allows a feasible constraint monitoring tool, such as a small size sensor platform. The multi-sensor device that we used in this work provided triaxial acceleration data and air pressure data. Multiple sensors embedded in a wearable device are more accurate for collecting different types of sensing information [23], but would be very inconvenient for the user. For this reason, we present only one single unit of sensor nodes, which collects multiple types of information.

Figure 2a shows a flow chart of our proposed system including four main components: i) the wearable sensor node to measure movement and height from ground by using a 3-axis accelerometer and a pressure sensor, ii) the wireless receiver to receive the measured data over Nordic wireless protocol and transmit it to PC over USB connection, iii) the activity monitor and analyzer working on the PC to aggregate the measured raw data and to analyze behavioral characteristics using features and classifiers, iv) the speaker to broadcast emergency alerts to their parents or a guardian.

As shown in Fig. 2b, the proposed activity recognition method using a 3-axis accelerometer and a pressure sensor comprises the following three steps: i) collecting and preprocessing the sensor data from an accelerometer, ii) extracting features, and iii) training and classification. In order to process, we have to remove the noise with a moving-average filter because errors made in the early steps may increase the classification error and the uncertainty on each further step. After preprocessing the sensors signals, it is necessary to choose the adequate features which we take as time-domain and frequency-domain features.

3. System overview

3.1. System architecture

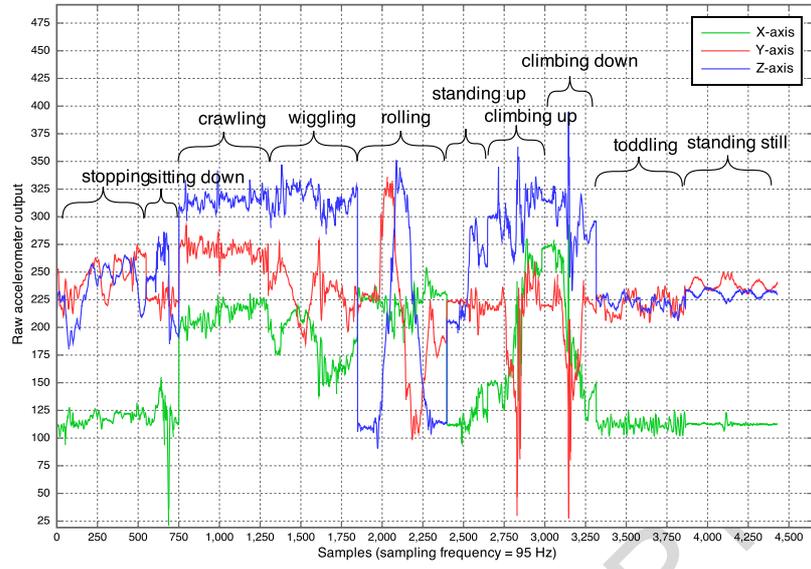
The developed wearable device basically consists of three layers: (i) communication layer, (ii) processing layer, and (iii) sensing layer. The communication layer enables ultra-low power wireless communication using a nRF24LE1 that is a dual-core communication chip of the wearable sensor device. The nRF24LE1 uses 2.4 GHz GFSK RF transceiver with embedded protocol engine, the Enhanced ShockBurst that enables data packet communication and supports autonomous protocol operation. This layer handles the transfer of collected and preprocessed data over the wireless network. The processing layer is where the preliminary activity recognition logic is applied to minimize network traffic and increase the lifespan of battery-powered nodes. This layer also manipulates raw data into a pre-defined data packet.

The sensing layer collects the context information such as child activities and object names around the child. In order to recognize various dangerous activities, we adopt multiple sensors as follows:

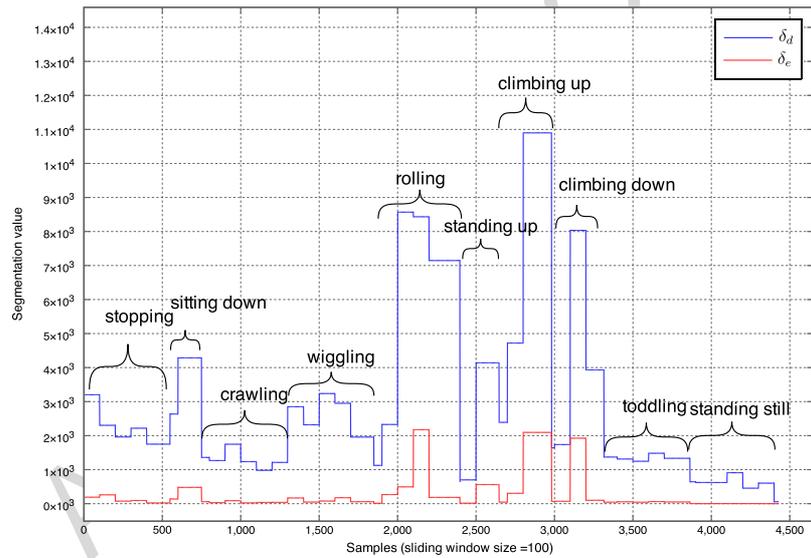
- A 3-axis accelerometer measures the movement, which consists of three signal-processing channels where it is low-pass filtered.
- A pressure sensor measures pressure enabling a measurement of distance between the grounds and the wearable sensor device that also could measure the temperature around the child. The pressure and temperature output data are calibrated and compensated internally.
- A RFID(SkyeModule M1-mini) is selected to read/write tags and smart labels, which has compatibility with most industry standard 13.56 MHz. This sensor allows us to recognize objects and space of a potentially dangerous situation.

The temperature sensor is internally equipped with the absolute pressure sensor for temperature compensation to measure more accurate pressure sensing. The temperature output data should be considered an important factor to detect different types of injuries in future work. However, the information is not utilized in our current research, since the scope of the paper is mainly focused on activity recognition.

From the data processing model, sensor fusion is grouped into three different levels of fusion which are direct data fusion, feature-level fusion, and decision-level fusion. If the sensors are measuring the same



(a) Raw 3-axis acceleration data



(b) Segmentation of acceleration values

Fig. 3. Raw data and segmentation of 3-axis acceleration data.

physical parameter with the data acquired commensurately, raw sensor data can be directly combined. Otherwise, the data needs to be fused at the feature or decision level. For feature-level fusion, features are first extracted from the sensor data to form a multi-dimensional feature vector so that general pattern recognition can be applied. We adopted this feature-level fusion to process our multi-sensor data from the wearable sensor device.

3.2. Data acquisition and segmentation

The embedded triaxial accelerometer can continuously sample the experienced accelerations at each sampling interval and produce 3-D acceleration readings, which are measures of the acceleration experienced in the three orthogonal axes: *X*-axis, *Y*-axis and *Z*-axis. Figure 3a shows the *X*-axis, *Y*-axis, and *Z*-axis readings for the different activities. The data for

our experiment was gathered in an experimental setting in which a sensor, with a sampling frequency of 95 Hz, was mounted at the waist of 3 healthy subjects of different age, weight, and height to collect eleven types of activity patterns. The time-domain features were computed from the most recent 255 samples and the frequency-domain features were computed from the same samples for efficient fast Fourier transform (FFT) implementation with 256 samples. We made the dataset available at <https://sites.google.com/site/yynams/baby-activity/>.

Let X , Y and Z denote the infinite data stream of measured acceleration values of the three space dimensions:

$$\begin{aligned} X &= (x_1, x_2, \dots), & Y &= (y_1, y_2, \dots), \\ Z &= (z_1, z_2, \dots). \end{aligned} \quad (1)$$

The corresponding data stream M of the magnitude is defined as:

$$\begin{aligned} M &= (m_1, m_2, \dots), \\ \text{where } m_i &= \sqrt{x_i^2 + y_i^2 + z_i^2}. \end{aligned} \quad (2)$$

The magnitude of the force vector is calculated by combining the measurements from all 3 axes using Eq. (2) to derive acceleration independent of orientation. A sliding window of the size n is pushed through the values of the incoming data stream with a specified offset. Suppose that $X' = (x'_1, \dots, x'_n)$ refers to the acceleration signals in the direction of the x -axis observed in the sliding window at a certain time. Y' , Z' and M' are defined analogously. A metric δ is applied to the data, which is then used to decide whether a state change occurred.

Basically, for segmentation of acceleration values, we used two segmentation approaches: energy-based segmentation as well as distance-based segmentation. The two segmentation approaches only differ in the type of metric that is applied. The energy-based approach uses as metric the variance of the magnitude values in the current sliding window, which can be determined as follows:

$$\begin{aligned} \delta_e(X', Y', Z', M') &= \text{var}(M') \\ &= \frac{1}{n} \sum_{i=1}^n (m'_i - \bar{m}')^2, \end{aligned} \quad (3)$$

where \bar{m}' is mean value of the magnitude values.

On the other hand, the distance-based approach uses as metric the distance of the pattern observed in the sliding window to a rest position. The rest position has to be recorded beforehand with time and is represented through the vector $(\bar{r}_x, \bar{r}_y, \bar{r}_z)$, where \bar{r}_x is mean value of the acceleration on the x -axis and so on. Then, the computation of the distance metric is as follows:

$$\begin{aligned} \delta_d(X', Y', Z', M') &= |(x'_1 - \bar{r}_x, \dots, x'_n - \bar{r}_x)| \\ &+ |(y'_1 - \bar{r}_y, \dots, y'_n - \bar{r}_y)| \\ &+ |(z'_1 - \bar{r}_z, \dots, z'_n - \bar{r}_z)|. \end{aligned} \quad (4)$$

Figure 3b shows examples of the energy-based segmentation and the distance-based segmentation from an accelerometer are plotted in Fig. 3a. However, it is difficult to discriminate different types of the dynamic activities based on the DC component, energy, and distance features. In the next section, we will describe methods to extract features from 3-axis acceleration data for discriminating between different activities.

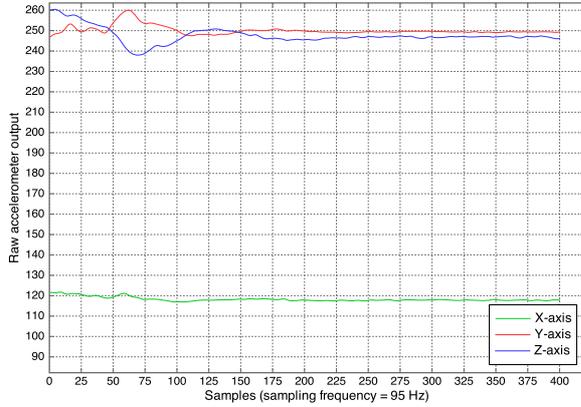
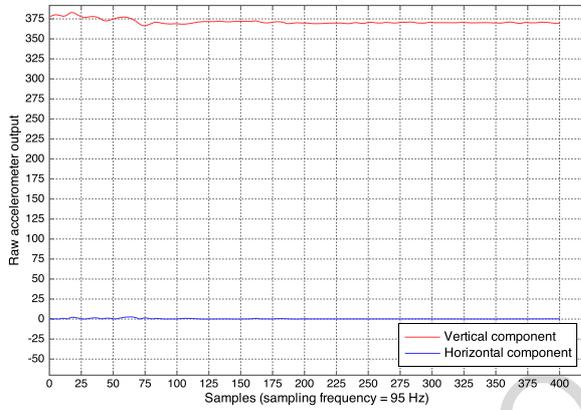
4. Feature extraction and activity recognition

4.1. Feature extraction

In general, signal feature could be analyzed in two domains; time-domain and frequency-domain. To extract features of the experimental data, we set window size to 256 and sample overlapping between consecutive windows to 128 at 95 Hz sampling frequency.

4.1.1. Time-domain feature

For time-domain features, each reading of accelerometer sensor consists of 3-D accelerations along X -axis, Y -axis and Z -axis according to local coordinate system of current orientation. For example, samples of (x, y, z) readings from an accelerometer are plotted in Fig. 4 while a baby sleeps on a bed. Basically, if the device is placed on a table, x and y should be zero since there is no acceleration at all, and z should be measured as $-g$ (where g represents acceleration due to gravity: 9.81 m/s^2). However, a baby does not lie straight on his/her back during sleeping. In addition, every time a baby breathes, sensor reading errors can affect data quality. There is a jittering noise in accelerometer data. We can reduce the effect of the jittering noise by scaling down (x', y', z') readings,

(a) Smoothed data of x -axis, y -axis, and z -axis

(b) Vertical and horizontal components

Fig. 4. Accelerometer data from a 3-axis accelerometer for sleeping.

followed by a smoothing technique using a moving-average filter $mv\text{-filter}$ of span L , as follows:

$$(x'', y'', z'') = mv\text{-filter}((x', y', z'), L). \quad (5)$$

The moving-average filter smooths data by replacing each data with the average of the neighboring data defined within L . The moving-average filter operates by averaging a number of points from the input signal to produce each point in the output signal. After selecting $L = 5$, a new series of readings are generated as shown in Fig. 4a. It can be seen that the proposed smoothing technique greatly removes the jittering noise and smooths out data by more significant standard deviation reduction than mean reduction. The effect of proposed smoothing effect on an accelerometer is shown in Fig. 4a. The de-noising process is important for building a recognition model using mean and standard deviation.

In [24,25], they have shown the signal average on each axis over a reasonable time period can produce a good estimate of the gravity-related component. We take a similar approach here to estimate the gravity component from each segment of (x', y', z') readings. Our estimation interval is set to the same as sample duration, which is to estimate the vertical acceleration vector \bar{v} corresponding to gravity. Therefore, $\bar{v} = (mx', my', mz')$, where mx' , my' , and mz' are means of respective axes for the sampling period. \bar{v} is normalized to \bar{v}_{norm} . Let $\bar{a}_i = (x'_i, y'_i, z'_i)$, $i = 1, 2, \dots, N$, be the vector at a given point in the sampling interval, where N is the length of sample points in the sample duration. The projection of \bar{a}_i onto the vertical axis \bar{v}_{norm} can be computed as the vertical component inside \bar{a}_i . Let p_i^{in} be the inner product and \bar{p}_i be the projection vector, as follows:

$$p_i^{in} = \langle \bar{a}_i, \bar{v}_{norm} \rangle, \quad (6a)$$

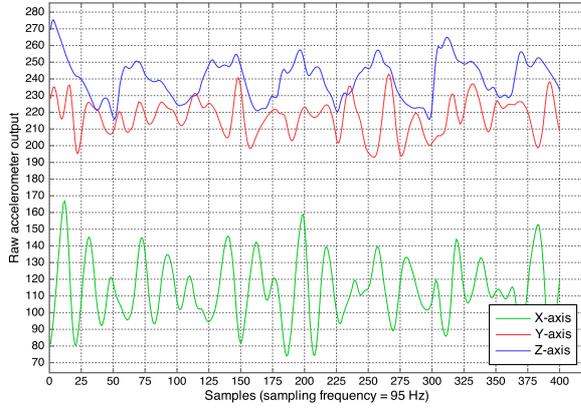
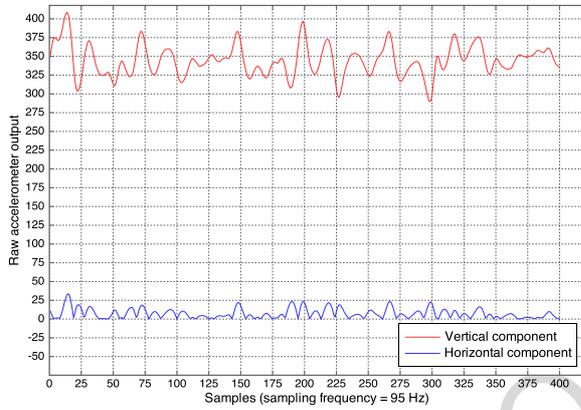
$$\bar{p}_i = p_i^{in} \cdot \bar{v}_{norm}, \quad (6b)$$

then the horizontal component \bar{h}_i of the acceleration vector \bar{a}_i can be computed as vector subtraction, as follows:

$$\bar{h}_i = \bar{a}_i - \bar{p}_i. \quad (7)$$

However, it is impossible to know the direction of \bar{h}_i relative to the horizontal axis in global 3-axis coordinate system. We only know \bar{h}_i lies in the horizontal plane that is orthogonal to estimated gravity vector \bar{v} . So we simply take the magnitude of \bar{h}_i , denoted by $\|\bar{h}_i\|$, as a measure of horizontal movement. The results of above algorithm generate two waveforms of $\{p_i^{in}, i = 1, 2, \dots, N\}$ and $\{\|\bar{h}_i\|, i = 1, 2, \dots, N\}$, which are amplitude of the vertical components and magnitude of the horizontal components, respectively. Each waveform is almost independent of orientation taking instant accelerometer samples. As illustrated in Fig. 4b, the vertical and horizontal components of accelerometer data in Fig. 4a are plotted according to above algorithm.

We mainly considered features including mean, standard deviation, energy, and correlation as Bao et al. [23] and Ravi et al. [26] selected. Four features were extracted from each of the three axes of the accelerometer as well as the vertical and the horizontal components, giving a total of nineteen attributes. The DC component of the signal over the window is the mean acceleration value. In order to show that activities

(a) Smoothed data of x -axis, y -axis, and z -axis

(b) Vertical and horizontal components

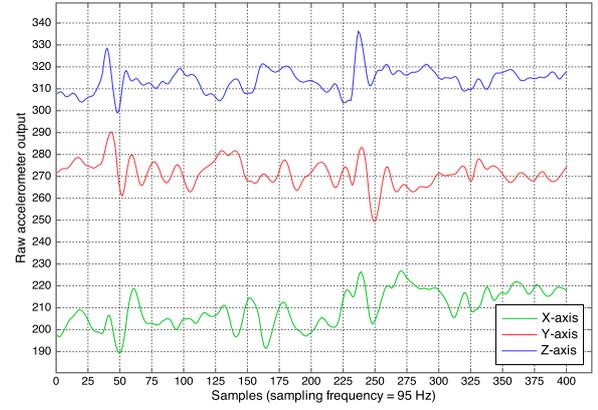
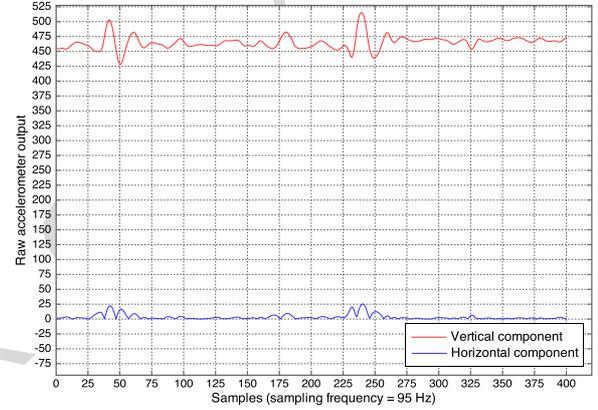
Fig. 5. Accelerometer data from a 3-axis accelerometer for walking.

cause the device to change orientation and this is easily detected from the accelerometer data, we present examples extracted from one baby. Standard deviation was used to capture the fact that the range of possible acceleration values differ for different activities such as walking and crawling as shown in Figs 5 and 6. In a comparison between walking and crawling, statistics of an accelerometer in walking, such as mean and standard deviation, are

$$\begin{aligned} & (m_x'', m_y'', m_z'', meanV, meanH) \\ & = (113.84, 217.38, 240.06, 344.30, 7.35) \end{aligned}$$

and

$$\begin{aligned} & (\sigma_{x''}, \sigma_{y''}, \sigma_{z''}, \sigma V, \sigma H) \\ & = (18.63, 10.55, 12.17, 20.42, 6.43). \end{aligned}$$

(a) Smoothed data of x -axis, y -axis, and z -axis

(b) Vertical and horizontal components

Fig. 6. Accelerometer data from a 3-axis accelerometer for crawling.

On the other hand, statistics of an accelerometer in crawling, such as mean and standard deviation, are

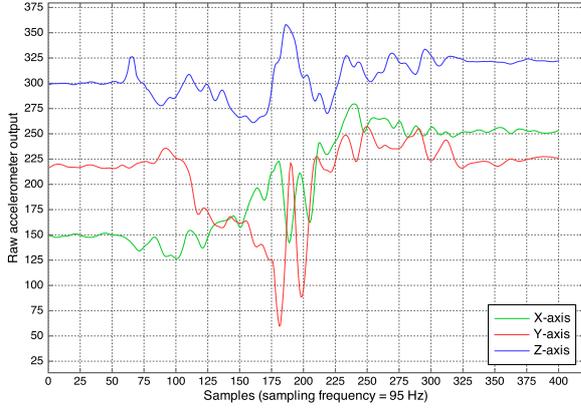
$$\begin{aligned} & (m_x'', m_y'', m_z'', meanV, meanH) \\ & = (209.50, 271.42, 313.88, 464.81, 3.41) \end{aligned}$$

and

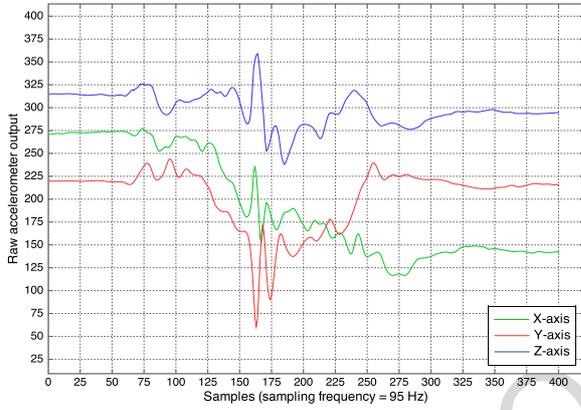
$$\begin{aligned} & (\sigma_{x''}, \sigma_{y''}, \sigma_{z''}, \sigma V, \sigma H) \\ & = (8.34, 5.81, 5.33, 11.01, 4.47). \end{aligned}$$

With respect to the smoothed accelerometer data and vertical components, standard deviation of walking is greater than that of crawling.

Figure 7 shows accelerometer data from a 3-axis accelerometer for climbing up and climbing down. The mean and standard deviation of an accelerometer in



(a) Climbing up



(b) Climbing down

Fig. 7. Accelerometer data from a 3-axis accelerometer for climbing up and climbing down.

climbing up are

$$(m_x'', m_y'', m_z'', meanV, meanH) = (204.01, 208.53, 306.26, 425.13, 5.41)$$

and

$$(\sigma_{x''}, \sigma_{y''}, \sigma_{z''}, \sigma V, \sigma H) = (50.63, 37.66, 19.42, 45.03, 7.46).$$

On the other hand, the mean and standard deviation of an accelerometer in climbing down are

$$(m_x'', m_y'', m_z'', meanV, meanH) = (193.63, 202.51, 298.47, 413.82, 4.94)$$

and

$$(\sigma_{x''}, \sigma_{y''}, \sigma_{z''}, \sigma V, \sigma H) = (56.93, 32.99, 18.15, 45.24, 7.76).$$

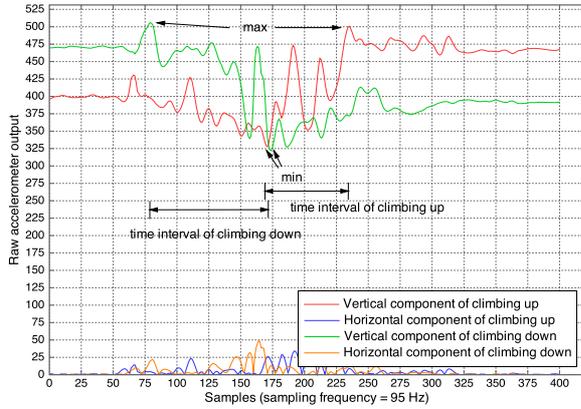
The feature characteristics of mean and standard deviation of climbing up are very similar to that of climbing down. In other words, it is difficult to discriminate between climbing up and climbing down based on mean and standard deviation features. In this paper, the slope mapping method is used to detect whether there is apparent fluctuation in the data series. We considered a simplified trunk tilt data series $i = 1$ to n . The slope filter is used to calculate the gradient over a specified window size. The apparent changes of slopes are investigated in vertical acceleration component. The slope s_i between two neighboring data window i and $i + 1$ is calculated and a segment slope series S with $n - 1$ data window is obtained from the original n -point data series. If the slope is equal to or greater than a predefined threshold of vertical acceleration average, the slope is calculated by

$$s_i = \begin{cases} \frac{V_{i+1}^{\max} - V_i^{\min}}{t_{i+1} - t_i}, & \text{if } meanV_i \leq meanV_{i+1} \\ & \text{and } |meanV_i - meanV_{i+1}| \geq meanV_{th} \\ \frac{V_{i+1}^{\min} - V_i^{\max}}{t_{i+1} - t_i}, & \text{if } meanV_i > meanV_{i+1} \\ & \text{and } |meanV_i - meanV_{i+1}| \geq meanV_{th} \\ 0, & \text{otherwise,} \end{cases} \quad (8)$$

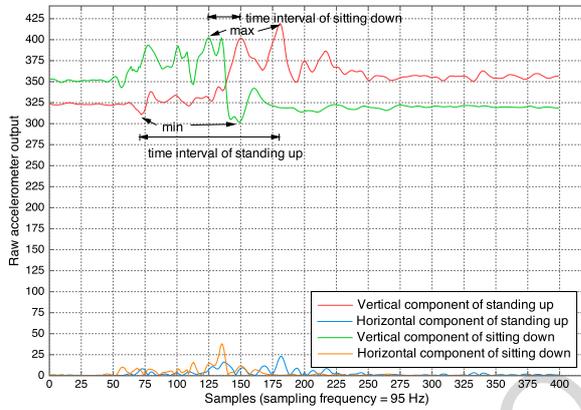
where V_i^{\max} and V_i^{\min} are the maximum and minimum of vertical acceleration values of data series i , respectively, $t_{i+1} - t_i$ is time interval between i and $i + 1$.

Figure 8a shows examples of the vertical and horizontal acceleration components of the climbing up and climbing down. For the climbing down case in this figure, it is observed that the maximum acceleration peak occurs prior to its minimum peak and vice versa for climbing up. This pattern exists in all samples collected in the test. Therefore, the rule is to compare the order of occurrences of those two peaks. If the maximum peak occurs prior to the minimum peak, this event could be a climbing down activity. On the contrary, if the occurrence of the maximum peak falls behind the minimum peak, the event could be a climbing up activity.

Furthermore, the time interval (or peak distance) between the maximum peak and the minimum peak is considered. The time interval rule is applied for clas-



(a) Climbing up and climbing down



(b) Standing up and sitting down

Fig. 8. Vertical and horizontal components for standing up, sitting down, climbing up, climbing down.

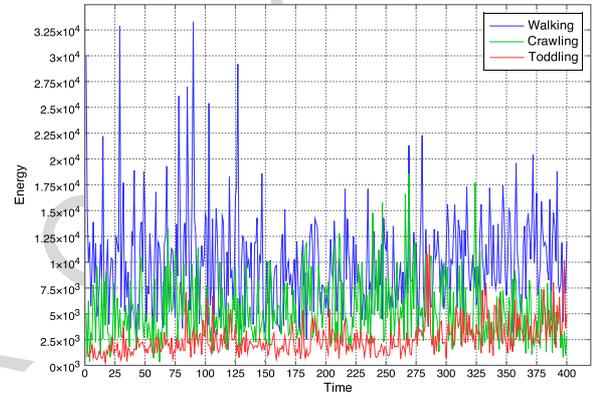
sification between standing up and sitting down activities. Figure 8b shows an example of vertical acceleration components during standing up and sitting down from the same young subject. The sitting down activity produces shorter peak distance while the standing up activity produces longer peak distance. For example, the patterns of standing up and sitting down shown in Fig. 8b have the peak distances in 109 and 24, respectively.

4.1.2. Frequency-domain feature

We computed the frequency-domain feature of 3-axis acceleration data as well as the vertical and horizontal components. As shown in Fig. 5, the periodicity in the data is reflected in the frequency domain. We mainly referred to successful FFT-based feature extraction [23,26,27] as a guideline for our implementation.



(a) Energy of vertical components



(b) Energy of horizontal components

Fig. 9. Energy of vertical and horizontal components for crawling, walking, and toddling.

To capture data periodicity, the energy feature was calculated. Energy E is the sum of the squared discrete FFT component magnitudes of the signal. The sum is divided by the window length w for normalization. If x_1, x_2, \dots are the FFT components of the window, then E is calculated by

$$E = \frac{\sum_{i=1}^w |x_i^2|}{|w|}. \quad (9)$$

Figure 9 shows examples of energy of vertical and horizontal components for crawling, walking, and toddling. In Fig. 9a, it can be seen that both E values of walking and toddling are around 1.2×10^7 , respectively while E values of crawling is around 2.2×10^7 . In the case of detecting walking and toddling, we found out that the vertical energy difference between those two states is small. However, the horizontal en-

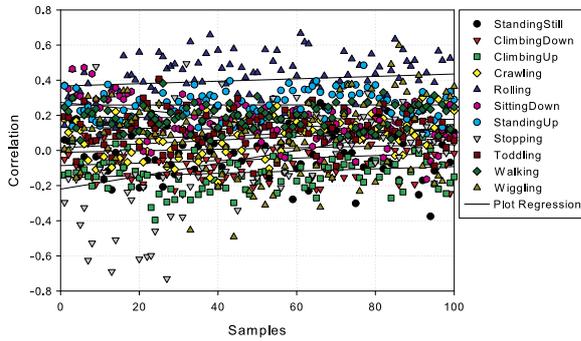


Fig. 10. Correlation between vertical and horizontal components for 11 activities.

ergy of toddling is different from that of walking. In case of toddling, a toddler walks sideways and backwards, runs well, falls, and stops easily. Thus, high peaks for the horizontal component, space out at highly periodic cycles as shown in Fig. 9b.

Correlation is especially useful for differentiating among activities that involve translation in just one dimension. For example, we can differentiate walking and running from stair climbing using correlation. Walking and running usually involve translation in one dimension whereas climbing involves translation in more than one dimension. Correlation ρ is calculated between each pair of axes as the ratio of the covariance and the product of the standard deviations by

$$\rho_{x,y} = \text{corr}(x,y) = \frac{\text{cov}(x,y)}{\sigma_x \sigma_y}. \quad (10)$$

Figure 10 shows results of correlation coefficient between vertical and horizontal components for 11 activities. In case of rolling, mean and standard deviations of correlation coefficient are 0.41 and 0.13. In the figure, the correlation coefficient of rolling is higher than that of other activities. In other words, the positive values of correlation coefficient indicate a stronger degree of linear relationship between vertical and horizontal variables such that as values for vertical component increases, values for horizontal component also increase.

4.1.3. Elevation feature

The pressure sensor is used for the determination of elevation. Pressure is measured by the miniaturized (diameter 6.1 mm, height 1.7 mm) SCP1000 Absolute Pressure Sensor from VTI Technologies. The air pressure sensor measures the atmospheric air pressure with a resolution of 1.5 Pa, which corresponds to about 10 cm at sea level. The data obtained from the pressure sensor needs to be normalized when both indoor

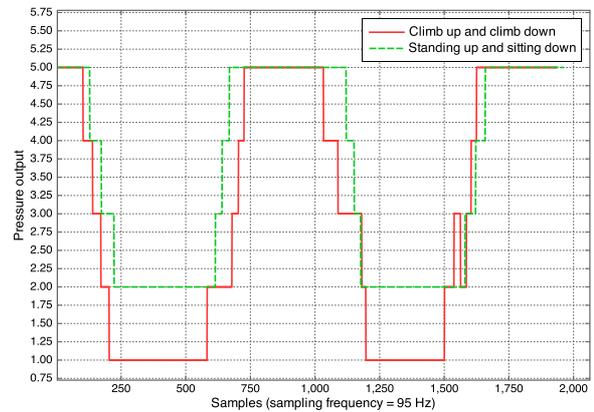


Fig. 11. Elevation data from an air pressure sensor for standing up, sitting down, climbing up, and climbing down.

and outdoor activity are analyzed. However, we limited the experiment to indoor activities. The elevation data obtained from the pressure sensor is useful to detect climbing up and climbing down events.

Figure 11 shows elevation data from an air pressure sensor for standing up, sitting down, climbing up, and climbing down. This absolute barometric pressure sensor gave a rough value for the altitude of the sensor. It was not perfect since in ideal conditions it can only resolve a vertical difference of 10 cm of air. The pressure data was converted to the height of the body-worn sensor from the ground. We configured pressure data of the ground as a reference value, and then the height was calculated by using the difference between the reference and measured value. In the figure, possible pressure values ranged from 1 to 5 for standing up and climbing up and climbing down, whereas standing up and sitting down activities lies within the range of 2–5. Using this range as our region of interest, we focus on classification of climbing up and down.

4.2. Activity recognition

After the feature extraction is completed, a classification procedure separates the child's activities from all other primitive features. The activity recognition algorithm should be able to recognize the accelerometer signal pattern corresponding to every activity.

4.2.1. Classification method using accelerometer data

We formulate activity recognition as a classification problem where classes correspond to activities and a test data instance is a set of acceleration values collected over a time interval and post-processed into a

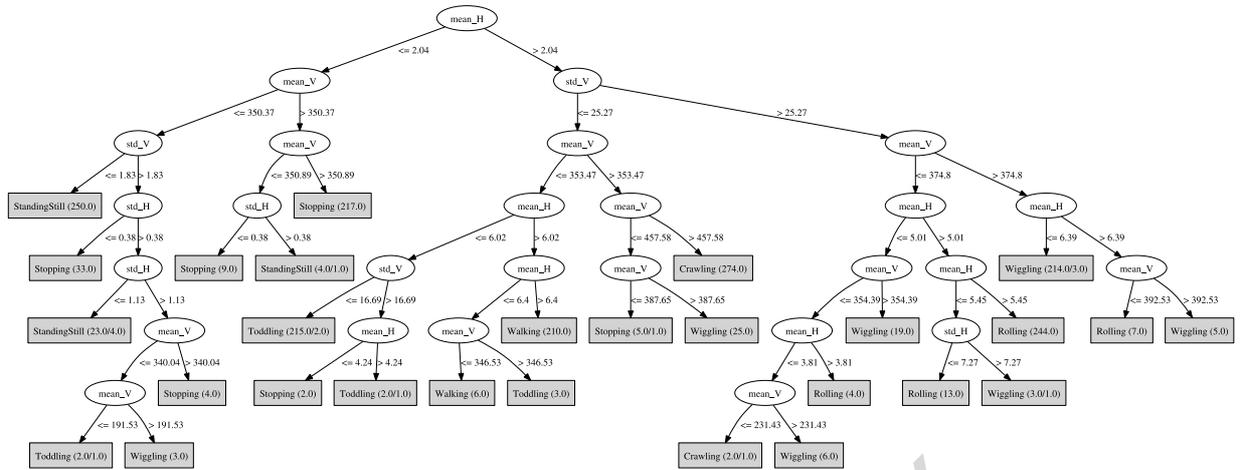


Fig. 12. Decision Tree generated from mean and standard deviation features of V/H.

single instance of mean, standard deviation, energy, correlation. In previous work to recognize activities, Decision Table [28], Decision Tree [29], Support Vector Machine (SVM) [30,31], and Nearest Neighbors [32], and Naive Bayes [33] classifiers were tested for activity recognition using the feature vector. Decision-based approaches [34] have been used in past work to recognize activities. Naive Bayes is a computationally efficient algorithm that has been used for pattern classification in a variety of applications. In addition, some studies have also focused on combining multiple types of sensors in addition to accelerometers for activity recognition.

In this paper, we evaluated the performance of Naive Bayes, Bayes Net, SVM, k -Nearest Neighbor (k -NN), Decision Table, Decision Tree, Multilayer Perceptron, and Logistic Regression classifiers, available in the Weka toolkit [35]. The Decision Tree learning method is one of the most widely used and practical techniques for inductive inference. The constructed tree first performs a binary split on the most salient feature (e.g. the X -axis acceleration energy from the sensor), dividing into two branches, then recursively constructs trees for each branch. The predictive values (e.g. walking, crawling, etc.) are assigned to the resulting leaves. To avoid over fitting to the data, which occurs after many tree subdivisions since each leaf then represents only a small number of samples, the tree is pruned. The Decision Tree classifier detects errors in classification of the training samples, as well as to errors in the attribute values of the samples, which provides a good balance between accuracy and computational complexity. A well-pruned Decision Tree model gen-

erated from simple V/H features for seven activities is presented in Fig. 12.

The purpose of distance-based clustering is to group large sets of data $S = \{u_i\}_{i=1}^N$ into clusters, each of which is represented by its mean $X = \{c_i\}_{i=1}^K$. Recently, the use of SVM has attracted significant research interest for accurate pattern classification as one of distance-based clustering methods [36]. An SVM model could represent aggregated data as points in space that is separated by optimal separating hyperplane. The 10-fold cross-validation is used to evaluate the SVM models. We put all the test cases in one dataset and then randomly divide it into 10 equal-sized pieces. Each time we choose one fold as the test dataset and the rest as the training dataset. We train the SVM model with the training dataset, evaluate it with the test dataset and get the precision p , recall r and the F -Measure weighted harmonic mean for each activity. After each fold is tested, we compute the average F -Measure of all the folds as the overall results for the activities. The definition is as following:

$$F - Measure = \frac{(\alpha + 1)rp}{r + \alpha p} \quad (11)$$

where α is the 'recall bias', a weighting of recall relative to precision declared in each scenario definition. In this paper, α is set to 1 for performance evaluation.

4.2.2. Classification method using elevation data

The pressure sensor signal is low-pass filtered using Butterworth filter and up-sampled using linear interpolation to the sampling rate of 95 Hz. The resulting signal is used to calculate the differential pressure

parameter ($\Delta P[k]$). The k th sample of ΔP signal is obtained considering the average pressure during T_w (we set to 2 seconds) before and T_w after each sample (overlapping the windows):

$$\Delta P[k] = \frac{T_s}{T_w} \left(\sum_{i=k}^{k+\frac{T_w}{T_s}} p[i] - \sum_{i=k-\frac{T_w}{T_s}}^k p[i] \right), \quad (12)$$

where $p[i]$ is the i th sample of the barometric signal and T_s is the sampling period. The ΔP signal is then normalized by dividing by the height of the subject.

If the thresholding of ΔP (with a heuristically determined threshold) indicates that the device altitude has significantly changed, the event is classified as standing up, sitting down, climbing up, and climbing down. If the system recognizes that no pressure change has occurred during T_{th} interval, then the classification is upgraded to non-moving status. If the system recognizes that the device altitude is significantly changed, even without detecting an activity based acceleration data, but the ΔP exceeds a high threshold ΔP_{th} , the event is classified as standing up, sitting down, climbing up, and climbing down. Although a 3-axis accelerometer is mainly used to detect activities including climbing up and climbing down, it is difficult to find the start and stop points of such activity in classifying it. In complementary sensor fusion, an event-driven reactive engine is adopted. We will discuss the event-driven reactive engine in the next section.

4.3. Event detection

In complementary fusion, each sensor captures different aspects and submits the information which is further merged. In cooperative fusion, sensors work together to gather complex information that is difficult to obtain from the sensors individually. In this paper, cooperative sensor fusion is applied to recognize activities such as climbing up and down. An accelerometer could only recognize the climbing activity as standing up and sitting down. A barometric sensor could only measure a variation of the height of the sensor from the ground. However, cooperative fusion of the two sensors enables recognition of the activity.

In order to achieve highly accurate results, we adopted an event-driven reactive engine based on an Event/Condition/Action (ECA) paradigm and constructed the decision rule about whether they can be a cause of home accidents or not. Depending on decision rule, the event detection method will signal an

alarm to give parents advance warning when the activity that can lead to an accident are detected. In this paper, the type of event is the result of the acceleration classification that mainly considered as a critical point of dangerous activity.

Conditions included the values of RFID tag, pressure, and temperature. The same condition may participate in more than one rule. We only considered an appropriate action as triggering alarm to the parents before the child faces dangerous situations. This in turn will alert and warn the parents in time to see what is happening with their babies and react to the situation immediately. The early warning system will give the parents enough time to save their babies and thus minimize any instances of falling accidents or sudden infant death syndrome (SIDS).

5. Experiments

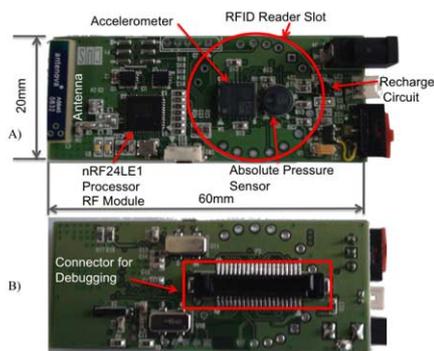
5.1. Implementation

We used multiple sensors embedded in a wearable device as shown in Table 1. We used the SCA3000 that is a 3-axis accelerometer for applications requiring high performance with low power consumption. It consists of three signal-processing channels where it is low-pass filtered and communicates with the processing layer is based on SPI bus that is a full duplex synchronous 4-wire serial interface. We also used the SCP1000 as a pressure sensor that measures absolute pressure to measure distance between the grounds and the sensor. The pressure and temperature output data are calibrated and compensated internally. The sensor communicates with the processing layer through SPI bus.

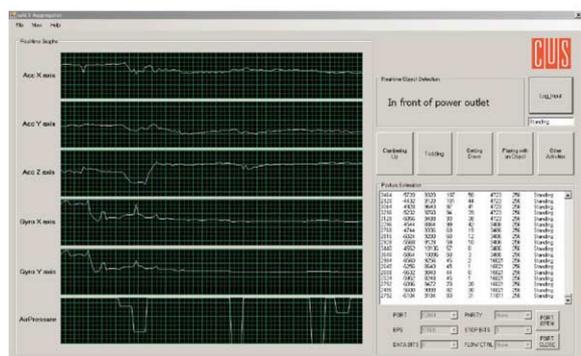
The SkyeModule M1-mini is selected to read and write tags, which has compatibility with most industry standard 13.56 MHz. It has a read/write distance that

Table 1
Multiple sensors embedded in a wearable device

Sensor type	Sensor name	Features
3-axis accelerometer	SCA3000	Measuring range is -2 g to $+2$ g and its sensitivity is 280 counts/g
Absolute pressure sensor	The SCP1000	Measuring range of the sensor is 30 kPa to 120 kPa
RFID	SkyeModule M1-mini	Compatibility with most industry standard 13.56 MHz such as ISO 15693, ISO 14443A, and ISO 18000-3



(a) Prototype of the wearable sensor device



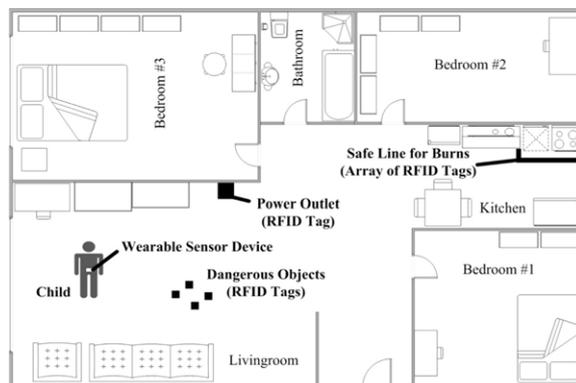
(b) Monitoring application

Fig. 13. Engineering prototype of the wearable sensor device and a screenshot of the monitoring application.

is typically greater than or equal to two inches for an ISO15693 RFID inlay. The sensor allows us to recognize objects and space that may cause dangerous situations. Finally, we developed the prototype wearable sensor device (size of 65 mm \times 25 mm) including the dual-core processor and sensors as shown in Fig. 13a. Figure 13b shows the developed monitoring application to aggregate sensing data and obtained annotations according to activities. It allows a visualization of sensing data in a real-time environment.

5.2. Experimental setup

Our observation was participatory with three volunteer families and conducted under the following conditions: i) the baby and child did not wear any devices and spent his/her leisure time with his/her mother, ii) the mother managed the housework as usual during the observation, iii) the experiments were conducted in 32.98 m² living room and 16.44 m² kitchen as shown in Fig. 14a. We recorded the experiments as videos that are synchronized with the monitoring application,



(a) Floor plan



(b) Wearable device

Fig. 14. Floor plan of an experimental environment and a child having the sensor device on the waist of the body.

and then annotated the raw data by comparing with the video. The experiments were performed in single-floor houses with one wearable sensor device for the child and the monitoring application operated on a laptop computer.

5.3. Experimental results

Accelerometer data was collected from three baby boys who are 16, 17 and 20 months old. There were total 1538 samples collected from one baby as training data, which are listed in Table 2. The other samples collected by the other baby were used as test dataset. We evaluated and compared several classifiers as mentioned in Section 4.2.1. 10-fold cross-validation was also used for testing. Different feature subsets are listed in Table 3. From the accuracy results, mean and standard deviation features contained more motion information than segmentation features because vertical and horizontal features can detect movement better than segmentation features. Although time-domain features are sufficient for different daily physical activities, the Decision Tree classifier using all features can better distinguish activities than that of using only

time-domain features. We found out that the results were substantially improved by using slope, energy, and correlation features, although the overall classification accuracy based on NB, BN, and SVM were not improved.

For a k -NN model ($k = 1$), when only $\{meanX, meanY, meanZ, stdX, stdY, stdZ, meanV, stdV, meanH, stdH, S_i, E, corrVH\}$ features were used, the accuracy was 95.5%. When only $\{meanX, meanY, meanZ, stdX, stdY, stdZ, meanV, stdV, meanH, stdH\}$ features were used, the accuracy was 93.9%. Compared to the accuracy results of mean and standard deviation features, there was at most 2.45% performance improvement with respect to k -NN, Decision Trees, MLP. Although MLP and Logistic achieve high recognition accuracy, these classifiers take 53.18 and 158.35 seconds to build a model. Table 4 shows precision and computational time of SVM depending on complexity parameters. SVM achieves high recognition accuracy when complexity parameter is 10000. However, the classifier takes 170.72 seconds to build a model. Thus, k -NN and Decision Trees are found to achieve high recognition accuracy with acceptable computational complexity. They are computationally

efficient, and their performance is suitable for real-time recognition. The recognition accuracy based on k -NN, J48, MLP, and Logistic classifiers were improved by slope, energy, and correlation features, even though the recognition accuracy based on NB and SVM classifiers were not improved.

Table 5 shows the detailed accuracy per classifier. The recall, or true positive rate (TP), is the proportion of correctly identified samples, the false positive rate (FP) is the proportion of incorrectly classified samples, and precision is the proportion of predicted positive cases. With respect to precision and recall, MLP, k -NN, and Decision Tree achieved accuracies above 94%. Table 6 demonstrates that in most cases we can achieve high levels of accuracy. For most common activities, we generally achieved accuracies above 90% based on k -NN, J48, and MLP. The classification between walking and toddling activities is more difficult than the classification between other activities. This can be partially explained by the similarity of these two activities and the basic characteristics of child activities such as unintended moving. Two activities may be considered the same activity in point of view, and can be considered only for moving detection. However, in this paper, two activities are classified by the horizontal energy. We achieve accuracies above 90% based on k -NN, J48, MLP, Logistic classifiers.

Crawling appears easier to identify than walking, which seems to make sense, since crawling involves more horizontal changes in acceleration. It appears much more difficult to identify climbing up and climbing down activities based on methods of excluding k -NN, J48, and MLP, but as we shall see shortly, that is because those two similar activities are often confused with one another. Our results indicate that none of the eight learning algorithms consistently performs best, but the multilayer perceptron does perform best overall. More detailed results are presented in Tables 7–14, which show the confusion

Table 2
Training dataset

Index	Label	# of samples
0	standing still	170
1	standing up	137
2	sitting down	136
3	walking	136
4	toddling	136
5	crawling	172
6	climbing up	137
7	climbing down	137
8	stopping	172
9	wiggling	171
10	rolling	171

Table 3

Feature subsets and recognition accuracy (NB: Naive Bayes, BN: Bayes Net, SVM: Support Vector Machine, k -NN: k-Nearest Neighbor, J48: Decision Tree, DT: Decision Table, MLP: Multilayer Perceptron, Logistic: Logistic Regression)

Feature set	NB	BN	SVM	k -NN	J48	MLP	Logistic
Segmentation features	34%	53.8%	46.3%	38.8%	50.2 %	47.7%	21%
Mean features	82.5%	78.6%	83.2%	87.9%	87.2 %	79.1%	83.7%
Std features	53.6%	60.1%	61.2%	73.8%	71 %	67.1%	58.5%
Mean and Std features	82.1%	86.6%	91.5%	93.9%	91.7%	91.5%	86.2%
Mean, Std, Slope, Energy, and Correlation features	81.2%	86.6%	93.8%	95.5%	93.9%	95%	88.7%
All features	81.7%	88.1%	95.2%	96.2%	94.7%	96.3%	93.2%

matrices associated with each of the eight learning algorithms.

The most important activities to analyze are the climbing up and climbing down activities. The confusion matrices indicate that many of the prediction errors are due to confusion between these two activities. If we focus on the results for the J48 decision tree model in Table 11, we see that when they are climbing up, the most common incorrect classification occurs when we predict climbing down, which occurs 30 times and accounts for a decrease in accuracy of 3.64% (20 errors out of 548). When the actual activity is climbing down, climbing up slightly outpaces wiggling in terms of the total number of errors (31 vs. 25).

Table 4
Precision and computational time of SVM

Complexity parameter	Precision	Time taken to build model (sec)
10^0	89.4%	0.83
10^1	92.2%	1.14
10^2	93.9%	3.65
10^3	94.9%	23.15
10^4	95.2%	170.72

Table 5
Detailed accuracy per classifier

Classifier	TP rate	FP rate	Precision	Recall	F-measure
NB	81.5%	1.8%	81.7%	81.5%	81.3%
BN	87.3%	1.12%	88.1%	87.3%	87.3%
SVM	95.2%	0.5%	95.2%	95.2%	95.2%
<i>k</i> -NN	96.2%	0.4%	96.2%	96.2%	96.2%
J48	94.7%	0.5%	94.7%	94.7%	94.7%
DT	74.9%	2.5%	75.8%	74.9%	74.7%
MLP	96.3%	0.4%	96.3%	96.3%	96.3%
Logistic	93.2%	0.6%	93.2%	93.2%	93.2%

On the other hand, using air pressure data, the activities of standing up, sitting down, climbing up, and climbing down have been recognized with more than 99% of accuracy as shown in Table 6. Such high accuracy is required for building safety applications based on for instance, falling and climbing up detection. Figure 15 illustrates clustering of the classes using classifier errors (J48); these were generated using the Weka machine learning explorer. It shows that most activities form separate clusters, while walking and toddling as well as climbing up and climbing down are closer together since these activities are very similar.

In addition to Fig. 15, Table 15 shows a summary of classifier errors (J48) including the mean absolute error, root mean squared error, relative absolute error, and root relative squared error. However, the most commonly reported errors are the mean absolute error and root mean squared error. In order to measure average model-performance errors, we chose the mean absolute error and root mean squared error as the standard error model in all the following analysis. Table 16 shows the error in accurately predicting class by different classifiers.

6. Conclusions

We have presented the activity recognition method for baby and child using only one triaxial accelerometer and a barometric sensor. We extracted time-domain and frequency-domain features for categorizing body postures such as standing still and wiggling as well as locomotion such as toddling and crawling. In order to improve the performance of the child activity recognition system, six features including magnitude, mean, standard deviation, slope, energy, and correlation are

Table 6

Recognition result comparison, a) standing still, b) standing up, c) sitting down, d) walking, e) toddling, f) crawling, g) climbing up, h) climbing down, i) stopping, j) wiggling, and k) rolling

Classifier	a	b	c	d	e	f	g	h	i	j	k
NB	85.2%	75.3%	78.1%	91.6%	92.4%	98.8%	57.5%	54.6%	76.5%	85.7%	94.9%
BN	86.6%	77.1%	90.4%	91.5%	91.5%	97.4%	77.3%	61.7%	97.5%	93.9%	96.3%
SVM	95.1%	97.2%	95.4%	96.9%	95.7%	99.3%	87.8%	85.5%	96.3%	95.9%	99.4%
<i>k</i> -NN	98.2%	95.3%	96.5%	92.4%	92%	98%	95.5%	94.5%	96.4%	97.5%	99.4%
J48	98.5%	93.3%	94%	93.3%	90.8%	98.7%	93%	89.4%	97.1%	92.7%	97.8%
DT	70.5%	66.7%	77.8%	80.7%	75.9%	76.8%	57.5%	56.2%	85.4%	83.7%	94.5%
MLP	97.2%	95.6%	94.7%	95.4%	94.6%	98.3%	96.5%	93.3%	97.1%	95.6%	99.3%
Logistic	95.1%	94.2%	94.6%	96.1%	95.3%	99.6%	82.3%	75.1%	95.5%	94.2%	99.3%
ΔP	–	99%	100%	–	–	–	99%	99%	–	–	–

Table 7

Confusion matrix for Naive Bayes, a) standing still, b) standing up, c) sitting down, d) walking, e) toddling, f) crawling, g) climbing up, h) climbing down, i) stopping, j) wiggling, and k) rolling

		Predicted class										
		a	b	c	d	e	f	g	h	i	j	k
Actual class	a	669	0	1	0	0	0	0	0	12	0	0
	b	0	277	104	0	4	0	0	17	147	0	0
	c	0	83	454	0	0	0	0	0	9	0	0
	d	0	0	3	513	30	0	0	0	0	0	0
	e	0	0	3	47	496	0	0	0	0	0	0
	f	0	0	0	0	0	678	2	0	0	9	0
	g	0	2	0	0	0	0	326	206	0	14	0
	h	0	6	0	0	0	0	154	343	0	42	3
	i	116	0	16	0	7	0	1	0	547	2	0
	j	0	0	0	0	0	8	84	60	0	499	33
	k	0	0	0	0	0	0	0	2	0	16	666
Precision	0.817	0.852	0.753	0.781	0.916	0.924	0.988	0.575	0.546	0.765	0.857	0.949

Table 8

Confusion matrix for Bayes Net

		Predicted class										
		a	b	c	d	e	f	g	h	i	j	k
Actual class	a	675	0	0	0	0	0	0	0	7	0	0
	b	0	482	36	0	4	0	0	20	7	0	0
	c	0	91	452	3	0	0	0	0	0	0	0
	d	0	1	2	508	35	0	0	0	0	0	0
	e	0	6	0	43	497	0	0	0	0	0	0
	f	0	0	0	0	0	680	2	0	0	7	0
	g	0	8	0	0	0	0	341	195	0	4	0
	h	0	6	0	0	0	0	87	439	0	14	2
	i	104	29	9	1	7	0	0	0	539	0	0
	j	0	2	1	0	0	18	11	55	0	573	24
	k	0	0	0	0	0	0	0	2	0	12	670
Precision	0.881	0.866	0.771	0.904	0.915	0.915	0.974	0.773	0.617	0.975	0.939	0.963

Table 9

Confusion matrix for SVM

		Predicted class										
		a	b	c	d	e	f	g	h	i	j	k
Actual class	a	659	0	1	0	0	0	0	0	22	0	0
	b	0	526	17	0	1	0	2	2	1	0	0
	c	0	7	537	0	0	0	0	0	2	0	0
	d	0	0	1	523	22	0	0	0	0	0	0
	e	0	1	0	16	529	0	0	0	0	0	0
	f	0	0	0	0	0	686	0	0	0	3	0
	g	0	2	1	0	1	0	477	64	0	3	0
	h	0	3	1	0	0	0	60	471	0	13	0
	i	34	2	5	1	0	0	0	0	647	0	0
	j	0	0	0	0	0	5	3	14	0	658	4
	k	0	0	0	0	0	0	1	0	0	9	674
Precision	0.952	0.951	0.972	0.954	0.969	0.957	0.993	0.878	0.855	0.963	0.959	0.994

Table 10
Confusion matrix for *k*-NN

		Predicted class										
		a	b	c	d	e	f	g	h	i	j	k
Actual class	a	660	0	0	0	1	0	0	0	21	0	0
	b	0	525	14	0	2	0	2	3	3	0	0
	c	0	13	531	0	1	0	0	0	1	0	0
	d	0	0	0	509	37	0	0	0	0	0	0
	e	0	0	0	41	505	0	0	0	0	0	0
	f	0	0	0	0	0	686	0	0	0	3	0
	g	0	2	1	0	0	0	529	14	0	2	0
	h	0	8	0	0	0	0	22	511	0	7	0
	i	12	3	4	1	3	0	0	0	666	0	0
	j	0	0	0	0	0	14	1	13	0	652	4
	k	0	0	0	0	0	0	0	0	0	5	679
Precision	0.962	0.982	0.953	0.965	0.924	0.92	0.98	0.955	0.945	0.964	0.975	0.994

Table 11
Confusion matrix for Decision Tree

		Predicted class										
		a	b	c	d	e	f	g	h	i	j	k
Actual class	a	671	0	0	0	1	0	0	0	10	0	0
	b	0	514	20	1	4	0	2	6	2	0	0
	c	0	21	519	3	1	0	0	0	2	0	0
	d	0	0	3	503	39	0	0	0	1	0	0
	e	0	4	0	32	505	0	0	0	5	0	0
	f	0	0	0	0	0	683	0	0	0	6	0
	g	0	5	3	0	0	0	503	30	0	6	1
	h	0	6	2	0	0	0	31	482	0	25	2
	i	10	1	5	0	6	0	0	1	666	0	0
	j	0	0	0	0	0	9	4	20	0	639	12
	k	0	0	0	0	0	0	1	0	0	13	670
Precision	0.947	0.985	0.933	0.94	0.933	0.908	0.987	0.93	0.894	0.971	0.927	0.978

Table 12
Confusion matrix for Decision Table

		Predicted class										
		a	b	c	d	e	f	g	h	i	j	k
Actual class	a	652	0	0	0	1	3	0	0	26	0	0
	b	15	396	76	4	17	15	5	8	10	0	3
	c	26	143	337	6	3	18	0	0	12	0	1
	d	15	6	2	413	93	15	0	0	2	0	0
	e	26	19	13	89	371	24	0	0	4	0	0
	f	4	0	0	0	0	685	0	0	0	0	0
	g	13	6	0	0	0	2	347	156	4	17	3
	h	9	11	0	0	0	4	188	289	9	36	2
	i	86	10	5	0	4	28	0	0	556	0	0
	j	44	0	0	0	0	49	58	60	19	431	23
	k	35	3	0	0	0	49	5	1	9	31	551
Precision	0.758	0.705	0.667	0.778	0.807	0.759	0.768	0.575	0.562	0.854	0.837	0.945

Table 13
Confusion matrix for Multilayer Perceptron

		Predicted class										
		a	b	c	d	e	f	g	h	i	j	k
Actual class	a	651	3	3	0	3	0	0	0	19	3	0
	b	0	524	15	0	4	0	1	5	0	0	0
	c	0	20	523	2	0	0	0	0	1	0	0
	d	0	0	2	524	20	0	0	0	0	0	0
	e	0	0	1	23	522	0	0	0	0	0	0
	f	0	0	0	0	0	685	1	0	0	3	0
	g	0	4	2	0	1	0	521	16	0	4	0
	h	0	6	0	0	0	1	13	516	0	12	0
	i	19	1	6	0	1	0	0	0	661	1	0
	j	0	0	1	1	0	11	3	16	0	647	5
	k	0	0	0	0	0	0	1	0	0	7	676
Precision	0.963	0.972	0.956	0.947	0.954	0.946	0.983	0.965	0.933	0.971	0.956	0.993

Table 14
Confusion matrix for Logistic regression

		Predicted class										
		a	b	c	d	e	f	g	h	i	j	k
Actual class	a	658	0	0	0	0	0	0	0	24	0	0
	b	0	519	22	0	2	0	1	3	2	0	0
	c	0	20	523	2	0	0	0	0	1	0	0
	d	0	0	0	523	23	0	0	0	0	0	0
	e	0	0	0	19	526	0	0	0	1	0	0
	f	0	0	0	0	0	687	0	0	0	2	0
	g	0	1	1	0	1	0	442	100	0	2	1
	h	0	6	1	0	0	0	90	423	0	28	0
	i	34	5	6	0	0	0	0	0	644	0	0
	j	0	0	0	0	0	3	4	37	2	634	4
	k	0	0	0	0	0	0	0	0	0	7	677
Precision	0.932	0.951	0.942	0.946	0.961	0.953	0.996	0.823	0.751	0.955	0.955	0.993

extracted from the preprocessed signals. We compared multiple feature sets to find an optimized classification method, and showed how well they performed on a body. We found that activities can be recognized with fairly high accuracy using a single triaxial accelerometer. In addition, we defined rules for a dangerous situation to utilize ECA-based reactive engine in the actual deployment in a home environment.

Using only a single wearable triaxial accelerometer sensor, the average overall accuracy of the k -NN and Decision Tree is 95.45% with acceptable computational complexity, which is better than we expected. Compared to only a single triaxial accelerometer sensor, our developed system, which included the pressure information, demonstrated an improved performance in detecting climbing up and down activities. Results

showed that using a barometric pressure sensor could reduce the incidence of false alarms. This work is significant because the activity recognition model permits us to gain useful knowledge about the habits of babies and children. Our work has a wide range of applications, including automatic customization of the mobile device's behavior based upon activities and generating a daily/weekly activity profile to determine if an obese child is performing a healthy amount of exercise. In future work, we plan to migrate the monitoring application into an embedded platform that can execute classification methods and ECA reactive engine at once. We will extend the application of the proposed system that will cover different floor levels by adding multiple bases and the function of air pressure during calibration. Furthermore, we will test how the

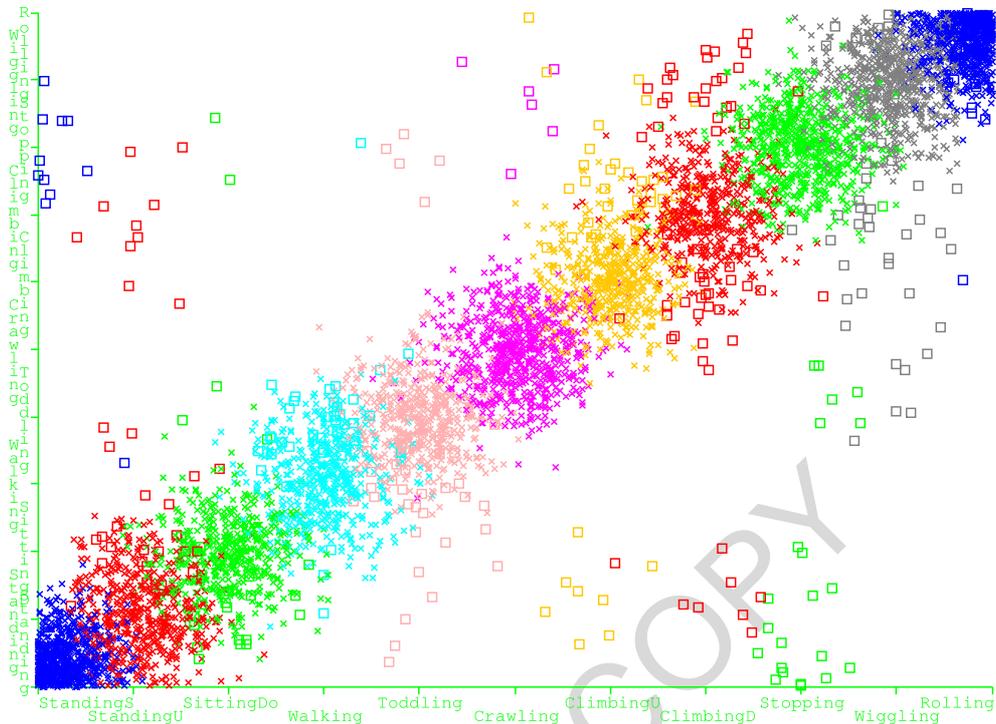


Fig. 15. Class clustering.

Table 15
Summary of classifier errors

	Predicted values
Correctly Classified Instances	6355 (94.6953%)
Incorrectly Classified Instances	356 (5.3047%)
Kappa statistic	0.9416
Mean absolute error	0.011
Root mean squared error	0.0956
Relative absolute error	6.6447%
Root relative squared error	33.2769%
Coverage of cases (0.95 level)	95.4999 %
Mean rel. region size (0.95 level)	9.4607%

Table 16

Mean absolute error and root mean squared error shown by different classifiers on predicting class

Classifier	Mean absolute error	Root mean squared error
NB	0.0339	0.1738
BN	0.0236	0.1421
SVM	0.1492	0.2642
k-NN	0.0073	0.0835
J48	0.011	0.0956
DT	0.0885	0.1904
MLP	0.0095	0.0745
Logistic	0.0208	0.0977

proposed method and system affect in daily life with its usability test.

Acknowledgment

The authors would like to thank all the volunteers who participated in our experiments. They would also like to thank Dianne Greco for previewing our manuscript. This research is supported by the International Collaborative R&D Program of the Ministry of Knowledge Economy (MKE), the Korean government, as a result of Development of Security Threat Control System with Multi-Sensor Integration and Image Analysis Project, 2010-TD-300802-002.

References

[1] M. Weiser, Ubiquitous computing, *Computer* **26** (Oct. 1993), 71–72.
 [2] K. Watanabe, Y. Kurihara, and H. Tanaka, Ubiquitous health monitoring at home – sensing of human biosignals on flooring, on tatami mat, in the bathtub, and in the lavatory, *Sensors Journal, IEEE* **9**(12) (Dec. 2009), 1847–1855.
 [3] J. Yin, Q. Yang, and J.J. Pan, Sensor-based abnormal human-activity detection, *IEEE Trans. on Knowl. and Data Eng.* **20** (Aug. 2008), 1082–1090.

- [4] T. Choudhury, G. Borriello, S. Consolvo, D. Haehnel, B. Harrison, B. Hemingway, J. Hightower, P.P. Klasnja, K. Koscher, A. LaMarca, J.A. Landay, L. LeGrand, J. Lester, A. Rahimi, A. Rea, and D. Wyatt, The mobile sensing platform: An embedded activity recognition system, *IEEE Pervasive Computing* **7** (Apr. 2008), 32–41.
- [5] H. Na, S. Qin, and D. Wright, A smart vision sensor for detecting risk factors of a toddler's fall in a home environment, in: *IEEE International Conference on Networking, Sensing and Control, 2007*, Apr. 2007, pp. 656–661.
- [6] H. Na, S. Qin, and D. Wright, Detecting fall risk factors for toddlers, *IEEE Pervasive Computing* **10**(1) (2011), 82–89.
- [7] Y. Nishida, Y. Motomura, K. Kitamura, and H. Mizoguchi, Infant behavior simulation based on an environmental model and a developmental behavior model, in: *IEEE International Conference on Systems, Man and Cybernetics, 2004*, Vol. 2, Oct. 2004, pp. 1555–1560.
- [8] N. Zouba, F. Bremond, and M. Thonnat, Multisensor fusion for monitoring elderly activities at home, in: *Proc. of the 2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*, ser. AVSS'09, IEEE Computer Society, Washington, DC, USA, 2009, pp. 98–103.
- [9] C. Zhu and W. Sheng, Multi-sensor fusion for human daily activity recognition in robot-assisted living, in: *Proc. of the 4th ACM/IEEE International Conference on Human Robot Interaction*, ser. HRI'09, ACM, New York, NY, USA, 2009, pp. 303–304.
- [10] R. Luo and M. Kay, Multisensor integration and fusion in intelligent systems, *IEEE Transactions on Systems, Man and Cybernetics* **19**(5) (Sep./Oct. 1989), 901–931.
- [11] R.R. Brooks and S.S. Iyengar, *Multi-Sensor Fusion: Fundamentals and Applications with Software*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1998.
- [12] R. Luo, C.-C. Yih, and K.L. Su, Multisensor fusion and integration: Approaches, applications, and future research directions, *Sensors Journal, IEEE* **2**(2) (Apr. 2002), 107–119.
- [13] U. Maurer, A. Smaligic, D.P. Siewiorek, and M. Deisher, Activity recognition and monitoring using multiple sensors on different body positions, in: *Proc. of the International Workshop on Wearable and Implantable Body Sensor Networks*, IEEE Computer Society, Washington, DC, USA, 2006, pp. 113–116.
- [14] A.Y. Yang, R. Jafari, S.S. Sastry, and R. Bajcsy, Distributed recognition of human actions using wearable motion sensor networks, *J. Ambient Intell. Smart Environ.* **1** (Apr. 2009), 103–115.
- [15] T.L.M. van Kasteren, G. Englebienne, and B.J.A. Kröse, Activity recognition using semi-markov models on real world smart home datasets, *J. Ambient Intell. Smart Environ.* **2** (Aug. 2010), 311–325.
- [16] Y. Cho, Y. Nam, Y.-J. Choi, and W.-D. Cho, Smartbuckle: Human activity recognition using a 3-axis accelerometer and a wearable camera, in: *Proc. of the 2nd International Workshop on Systems and Networking Support for Health Care and Assisted Living Environments*, ser. HealthNet'08, ACM, New York, NY, USA, 2008, pp. 7:1–7:3.
- [17] E. Choi, Y. Nam, B. Kim, and W. Cho, An incremental statistical method for daily activity pattern extraction and user intention inference, *TIIS* **3**(3) (2009), 219–234.
- [18] N.C. for Health Statistics (U.S.), Injury in the United States: 2007 Chartbook, ser. DHHS publication, U.S. Department of Health and Human Services, Centers for Disease Control and Prevention, National Center for Health Statistics, 2008.
- [19] G.B.C.A. Directorate, Research on the pattern and trends in home accidents: Government consumer safety research, The Department, 1999.
- [20] M. Kim, A survey on children accidents (written by Korean), Survey reports of Korea Consumer Agency, Apr. 2007.
- [21] Factsheet of child accidents facts, Survey reports of CAPT (Child Accident Prevention Trust) UK, Apr. 2008.
- [22] S.C.A. Thomopoulos, Sensor integration and data fusion, *Journal of Robotics Systems* **7** (1990), 337–372.
- [23] L. Bao and S.S. Intille, Activity recognition from user-annotated acceleration data, *Pervasive Computing* (2004), 1–17.
- [24] D. Mizell, Using gravity to estimate accelerometer orientation, in: *Proc. of the 7th IEEE International Symposium on Wearable Computers*, ser. ISWC'03, IEEE Computer Society, Washington, DC, USA, 2003, pp. 252–253.
- [25] J. Yang, Toward physical activity diary: Motion recognition using simple acceleration features with mobile phones, in: *Proc. of the 1st International Workshop on Interactive Multimedia for Consumer Electronics*, ser. IMCE'09, ACM, New York, NY, USA, 2009, pp. 1–10.
- [26] N. Ravi, N. Dandekar, P. Mysore, and M.L. Littman, Activity recognition from accelerometer data, in: *Proc. of the 17th Conference on Innovative Applications of Artificial Intelligence*, Vol. 3, AAAI Press, 2005, pp. 1541–1546.
- [27] B. Baas and S.U.D. of Electrical Engineering, An approach to low-power, high-performance, Fast Fourier Transform processor design, Stanford University, 1999.
- [28] R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, in: *Proc. of the 14th International Joint Conference on Artificial Intelligence*, Vol. 2, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1995, pp. 1137–1143.
- [29] J.R. Quinlan, *C4.5: Programs for Machine Learning*, (Morgan Kaufmann Series in Machine Learning), 1st edn, Morgan Kaufmann, Oct. 1992.
- [30] J.C. Platt, Fast training of support vector machines using sequential minimal optimization, in: *Advances in Kernel Methods*, B. Schölkopf, C.J.C. Burges, and A.J. Smola, eds, MIT Press, Cambridge, MA, USA, 1999, pp. 185–208.
- [31] S.S. Keerthi, S.K. Shevade, C. Bhattacharyya, and K.R.K. Murthy, Improvements to platt's smo algorithm for svm classifier design, *Neural Comput.* **13**(3) (Mar. 2001), 637–649.
- [32] F. Foerster, M. Smeja, and J. Fahrenberg, Detection of posture and motion by accelerometry: A validation study in ambulatory monitoring, *Computers in Human Behavior* **15**(5) (Sep. 1999), 571–583.
- [33] G.H. John and P. Langley, *Estimating Continuous Distributions in Bayesian Classifiers*, 1995, pp. 338–345.
- [34] K. Aminian, P. Robert, E. Buchser, B. Rutschmann, D. Hayoz, and M. Depairon, Physical activity monitoring based on accelerometry: Validation and comparison with video observation, *Medical and Biological Engineering and Computing* **37**(3) (May 1999), 304–308.

- [35] I.H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd edn, Morgan Kaufmann Series in Data Management Systems, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2005.
- [36] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines: And Other Kernel-Based Learning Methods*, Cambridge University Press, New York, NY, USA, 2000.

AUTHOR COPY