Efficient Activity Recognition and Fall Detection Using Accelerometers

Simon Kozina, Hristijan Gjoreski, Matjaž Gams, Mitja Luštrek

Department of Intelligent Systems, Jožef Stefan Institute
Jamova cesta 39, 1000 Ljubljana, Slovenia
{simon.kozina, hristijan.gjoreski, matjaz.gams, mitja.lustrek}@ijs.si

Abstract. Ambient assisted living (AAL) systems need to understand the user's situation, which makes activity recognition an important component. Falls are one of the most critical problems of the elderly, so AAL systems often incorporate fall detection. We present an activity recognition (AR) and fall detection (FD) system aiming to provide robust real-time performance. It uses two wearable accelerometers, since this is probably the most mature technology for such purpose. For the AR, we developed an architecture that combines rules to recognize postures, which ensures that the behavior of the system is predictable and robust, and classifiers trained with machine learning algorithms, which provide maximum accuracy in the cases that cannot be handled by the rules. For the FD, rules are used that take into account high accelerations associated with falls and the recognized horizontal orientation (e.g., falling is often followed by lying). The system was tested on a dataset containing a wide range of activities, two different types of falls and two events easily mistaken for falls. The F-measure of the AR was 99 %, even though it was never tested on the same persons it was trained on. The F-measure of the FD was 78 % due to the difficulty of the events to be recognized and the need for real-time performance, which made it impossible to rely on the recognition of long lying after a fall.

Key words: Ambient assisted living, Activity recognition, Fall detection, Machine learning, Rules, Accelerometers

1   Introduction

The world's population is aging rapidly, threatening to overwhelm the society's capacity to take care of its elderly members. The percentage of persons aged 65 or over in developed countries is projected to rise from 7.5% in 2009 to 16% in 2050 [1]. This is driving the development of innovative ambient assisted living (AAL) technologies to help the elderly live independently for longer and with minimal support from the working-age population [2, 3].

This paper presents a system that recognizes the user's activity and detects falls using wearable sensors. To provide timely and appropriate assistance, AAL systems must understand the user's situation and context, making activity recognition (AR) an essential component [4, 5]. Fall detection (FD) is an important component of many

AAL systems because approximately half of the hospitalizations of the elderly are caused by falls [6], fear of falling is an important cause for nursing home admission [7], and "the long lie" (not being able to get up and call for help) is a good predictor of death within six months [8].

AR and FD require a sensor system that observes the user and intelligent software that infers the user's activities from the sensor data [9, 10]. We selected wearable accelerometers as the sensors since they are accurate, inexpensive, can be used both indoors and outdoors, and can be sewn into clothing for minimal intrusiveness. For the AR, we developed an architecture that combines rules to recognize postures, which ensure the behavior of the system is predictable and robust, and classifiers trained with machine learning (ML) algorithms, which provide maximum accuracy in the cases that cannot be handled by the rules. For the FD, rules are used that take into account high accelerations associated with falls and the recognized horizontal orientation (e.g., falling is often followed by lying).

The rest of this paper is organized as follows. An overview of studies related to AR and FD is presented in Section 2. Section 3 gives an overview of the system and the methodology. In Section 4, the sensors and methods used in the AR and FD are described. Section 5 and 6 describe the experimental setup and experimental results, respectively. Section 7 concludes the paper and gives directions for future work.

2 Related Work

AR and FD approaches can be divided into those that use wearable and non-wearable sensors, respectively. The most common non-wearable approach is based on cameras [11]. Although this approach is physically less intrusive for the user compared to one based on wearable sensors, it suffers from problems such as target occlusion, time-consuming processing and privacy concerns. The most mature approach to both AR and FD is probably using wearable accelerometers [12, 13, 14, 15]. There are two common types of wearable-sensor approach that have proven successful: those that use domain knowledge encoded with rules, and those that use machine learning. Most researchers used only one of the two approaches, while our work combines both.

The most common accelerometer-based AR approach uses only ML. Typically a sliding window passes over the stream of sensor data, and the data in each window is classified with one of the known classification methods, such as decision trees (DTs) and support vector machines (SVM). Examples include Kwapisz et al. [12], who used an accelerometer placed on the thigh and compared three classification methods on dynamic activities such as walking, running, and jogging. Ravi et al. [16] used an accelerometer in a mobile phone and tested five classification methods. The results showed that when a given person's data was used for both training and testing, the accuracy was 90%, but when a different person's data was used for the testing, the accuracy dropped to 65%. In our work we never used the same person for training and testing, since the developed model is intended for use by people who were not involved in the training.

An alternative approach to accelerometer-based AR is based on manually created rules. These rules are usually based on features that are calculated from sensor orientations and accelerations. Wu et al. [13] presented an approach in which decision rules are used to recognize activities. Another implementation of such rules was presented by Lai et al. [14], who used six accelerometers, placed on the neck, waist, both wrists and both thighs. The reported accuracy was almost perfect (99.5%), but the number of sensors is excessive for everyday use.

The most common approach to FD use rules that apply thresholds to accelerations and features derived from them, and sometimes consider the activity after a potential fall. Jantaraprim et al. [17] used a triaxial accelerometer worn on the chest; by applying a simple threshold to the acceleration, they detected falls with 98.9 % accuracy. Nguyen, Cho and Lee [19] used a triaxial accelerometer worn on the waist; by applying thresholds to the acceleration, they detected a potential fall and the activity after the fall, resulting in 100 % accurate fall detection. Some researchers used machine learning instead of threshold-based algorithms. Zhang et al. [20] and Shan and Yuan [18] both used a triaxial accelerometer worn on the waist. Using SVM machine learning algorithm on various features derived from accelerations, they detected falls with 96.7 % and 100 % accuracy, respectively. We opted for the rules-based approach, since rules can be understood by humans and are thus less likely to result in unexpected behavior in practice.

3   System implementation

An overview of the system is shown in Fig. 1. Two accelerometers were attached to the user's chest and thigh. The placement was chosen as a trade-off between the physical intrusiveness and the performance in preliminary tests [11]. The Shimmer sensor platform [21] was chosen because it has a reasonable battery life and compact size, is completely wireless, and has the option to reprogram the sensor based on the user's needs and situation. The platform has a 3-axis accelerometer, uses Bluetooth communication, and has 2 GB of storage, which is enough to store 3 months of sensor data when the frequency of acquisition is 50 Hz. This frequency proved sufficient to capture even the fastest human movement. Additionally, for the purpose of the EvAAL AR competition, a laptop with a long-range Bluetooth antenna will be used for maximum reliability. In general, though, any kind of Bluetooth device with modest processing capability is sufficient, therefore a smartphone can also be used.
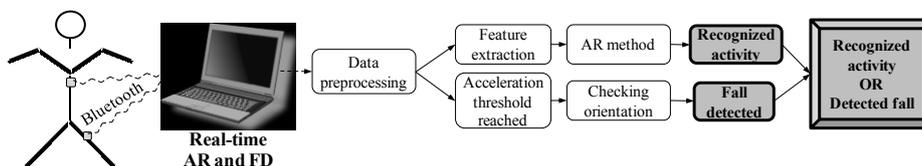

Fig. 1. System overview.

The process of the AR and FD is the following. First, the sensors transmit the raw acceleration data over Bluetooth to the processing unit. The data from both sensors is then synchronized, filtered and segmented. Then the flow splits into two. On one side the segmented data is transformed into feature vectors for the AR module, which recognizes the user's activity. On the other side, the FD module checks the acceleration pattern. If a fall pattern is recognized and the predefined acceleration threshold is reached, the user's orientation is checked. If the orientation corresponds to lying (horizontal), a fall is detected. Both the AR and FD modules are evaluating the user's situation every 250 milliseconds which is chosen with accordance to the EvAAL AR competition guidelines. First, the FD module checks for the fall acceleration pattern. If a fall is detected, the system outputs the activity falling; otherwise the AR recognizes and outputs the activity. For instance, if the current system time is denoted with t, the FD module evaluates fall events in the $[t - 2 \text{ s}, t - 1 \text{ s}]$ interval, and the $[t - 1 \text{ s}, t]$ interval is used to check if the orientation of the chest sensor corresponds to lying. If the fall event is detected and the orientation is correct, the reported activity is falling, otherwise the reported activity is computed with the AR module in the $[t - 2 \text{ s}, t]$ interval. The system thus reports the user's activity with a two-second delay.

4 Methods

4.1 Data Preprocessing

The first step in the preprocessing phase is sensor data synchronization. This is necessary when multiple sensors are used, since the data from the sensors is not all received at the same time.

Once the sensor measurements are synchronized, further preprocessing is performed using band-pass and low-pass filters. The acceleration is the sum of the acceleration due to the gravity and the acceleration due to the movement of the sensor (and the person wearing it). The band-pass filter thus has two goals: (1) to eliminate the low-frequency acceleration (gravity) that captures information about the orientation of the sensor with respect to the ground and (2) to eliminate the high-frequency signal components generated by noise, thus preserving the medium-frequency signal components generated by dynamic human motion. The band-pass-filtered data is used for the extraction of features relevant for dynamic activities, such as walking and cycling. The low-pass filter is used to eliminate most of the signals generated by dynamic human motion, preserving the low-frequency component, i.e., gravity [22]. The low-pass-filtered data thus contains sensor orientation information, which is relevant for the recognition of static activities (postures), such as lying, sitting and standing.

Finally, an overlapping sliding-window technique is applied. A window of 2-second size (width) moves across the stream of data, advancing by 250 milliseconds in each step. The window size was selected in correspondence to the EvAAL AR competition guidelines. The data within each window is used to compute the feature vector used for AR described in the next section. The feature vector contains low-pass-filtered features that measure the posture of the body. Additionally, it contains

band-pass-filtered features that represent: (1) the motion shape, (2) the motion variance, (3) the motion energy, and (4) the motion periodicity [22]. The feature vector consists of a total of 64 features per sensor.

4.2   Activity Recognition Module

In the AR module, the activities are recognized by a three-level scheme shown in Fig. 2 [23]. On the first level the feature vector is fed into a Random Forest classifier, which is trained to distinguish cycling from the other activities. If an activity is not classified as cycling, the feature vector is passed to the second level, where the activities are recognized by rule-based activity recognition (R-BAR). On this level only features representing average values of low-pass-filtered data are used. The following activities are separated at this level: sitting, lying, bending, and upright posture. If the selected activity is an upright posture, the third level of activity recognition is used to distinguish between standing and walking. The feature vector is again fed into a Random Forest classifier, which is trained to separate these activities. The parameters used for the Random Forest classifier were default ones, as described in Weka's API [24].
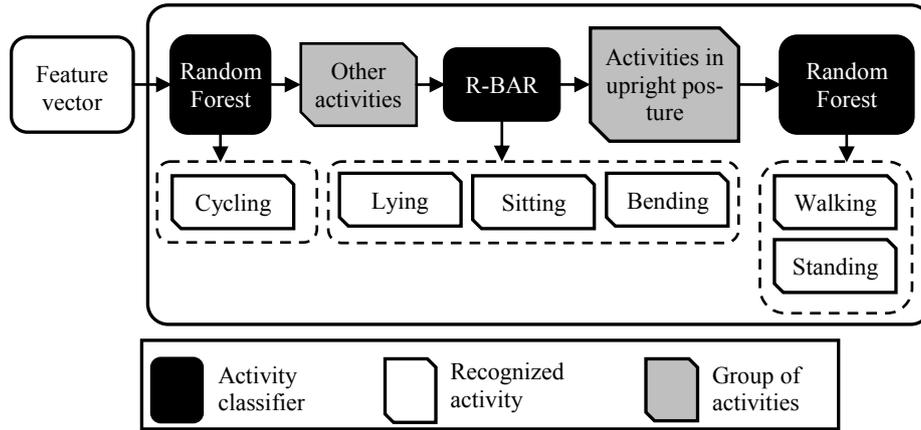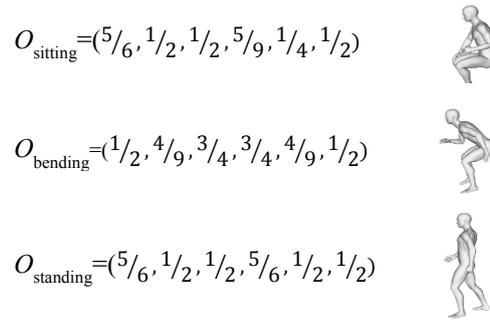

Fig. 2. A three- level AR scheme.

R-BAR, used on the second level of the AR, is used for detecting static activities, standing, lying, sitting and bending. Walking, which is a dynamic activity, is merged with its static equivalent, standing. R-BAR uses the orientation of the sensors to recognize posture. The orientation of a sensor is computed with Eq. (1), where i is one of the axes (x, y, or z). Additionally, the orientation is normalized to [0,1] interval.

$$\phi_i = \left(arccos \left(\frac{a_i}{\sqrt{a_x^2 + a_y^2 + a_z^2}} \right) + 1 \right) \cdot \frac{1}{2} \tag{1}$$

The values computed in this way form an orientation vector $O=(\phi_x, \phi_y, \phi_z)$ for one sensor, which is then matched with the set of rules defined by a domain expert as the typical orientations of the sensors for each activity. Figure 3 shows example orientations for three activities (sitting, bending, and upright posture). The structure of the rules in Fig. 3 is the following: $O_{activity}=(\phi_{chest,x}, \phi_{chest,y}, \phi_{chest,z}, \phi_{thigh,x}, \phi_{thigh,y}, \phi_{thigh,z})$. For every orientation measurement in vector O, an error is computed using Eq. (2), where d is the absolute difference between the value defined in the rules and the actual measurement. A higher absolute difference d denotes a higher difference between the actual and the typical sensor orientation, resulting in a larger value of the error e.

$$e = \begin{cases} \dfrac{d^4}{0.25^3}; & 0 \leq d < 0.25 \\ 3d - 0.5; & 0.25 \leq d < 0.5 \\ 1; & 0.5 \leq d \end{cases} \tag{2}$$

The error values form an error vector whose size is the same as that of the orientation vector. These components are summed up in order to obtain the overall error of an activity. Activity with the minimum error is selected as the correct one.

$O_{sitting}=(^5/_6, ^1/_2, ^1/_2, ^5/_9, ^1/_4, ^1/_2)$

$O_{bending}=(^1/_2, ^4/_9, ^3/_4, ^3/_4, ^4/_9, ^1/_2)$

$O_{standing}=(^5/_6, ^1/_2, ^1/_2, ^5/_6, ^1/_2, ^1/_2)$

Fig. 3. Example orientation for three activities: sitting, bending and standing.

4.3 Fall Detection Module

A typical acceleration pattern during a fall, measured by a chest accelerometer, is a decrease in acceleration followed by an increase, as shown in Fig. 4. This is because an accelerometer at rest registers 1 g (the Earth's gravity) and during free fall 0 g. When a person starts falling, the acceleration decreases from 1 g to around 0.5 g (perfect free fall is never achieved). Upon the impact with the ground, a short strong increase in the acceleration is measured.
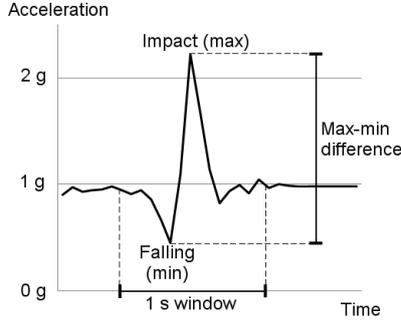
Fig. 4. Acceleration pattern during a fall.

To detect falls with a threshold, we used the length of the acceleration vector, which means that we ignored the direction of the acceleration. The minimum and the maximum acceleration within a one-second window were measured. If the difference between the maximum and the minimum exceeded 1 g and the maximum came after the minimum, we declared that a fall had occurred.

We augmented fall detection with the measurement of the person's orientation after a potential fall. We assumed that the acceleration vector $a = [a_x, a_y, a_z]$, which consists of the accelerations along the three axes of the accelerometer, generally points downwards (in the direction of the Earth's gravity). Let z be the axis pointing downwards when the person is standing upright. The angle φ between the acceleration vector and the z axis thus indicates the person's orientation, and was computed as follows:

$$cos\varphi = \frac{a_z}{\sqrt{a_x^2 + a_y^2 + a_z^2}} \tag{3}$$

A person was considered to be oriented upright if $-35° < \varphi < 35°$. This was used for fall detection: if an acceleration fall pattern was detected that exceeded the threshold as described previously, and the orientation in the next second was not upright, we declared that a fall had occurred.

5 Experiments

5.1 Test Scenario

In order to evaluate the AR and FD methods, a complex, 90-minute, scenario was designed in cooperation with a medical expert to capture the real-life conditions of a person's behavior, although it was recorded in a laboratory. The scenario contained several sub-scenarios: walking on a treadmill, cycling on a stationary bicycle, elementary activities such as: sitting, lying, standing, and specialized activities such as: cooking, reading, typing, washing dishes, and scrubbing the floor. A special "fall" sub-scenario was included in order to evaluate the FD method. It contained two non-fall-like events with large accelerations (quickly sitting down and quickly lying down) and two fall events – tripping and falling slowly (fainting).

5.2 Evaluation metrics

The AR and FD methods were experimentally evaluated on the described test scenario. The evaluation technique for the ML methods, the ones that require training a model, was leave-one-person-out cross validation. This means that a model was trained on the recordings of all the people except one. The remaining person was used to evaluate the model. This procedure was repeated for each person (10 times) and the average performance was measured. This evaluation technique was used because training and testing on the same person's recordings would give overly optimistic results if the intended use of the model is to classify the activities of previously unseen people.

When selecting the measure of performance, we had to decide how to weigh the activities with different total durations, and even more importantly, how to weigh undetected falls compared to falsely detected falls. Both are important: not detecting a fall may endanger a person's health, while false alarms make the system unlikely to be used in real life. We eventually decided to follow the example of the EvAAL AR competition and use the F-measure both for AR and FD, since it weighs all the activities, as well as undetected and falsely detected falls equally. It is defined as a harmonic mean of recall (the percentage of the events recognized as falls/non-falls from all the fall/non-fall events) and precision (the percentage of the events truly being falls/non-falls of all the events recognized as such).

6 Results and Discussion

6.1 Activity Recognition

Table 1 shows the comparison between AR module designed for EvAAL competition and other standard machine learning methods. Using F-measure for the performance evaluator, the AR module outperforms other standard methods from 0.79 percentage points (Random Forest) up to 2.68 percentage points (Decision Tree).

Table 1. EvAAL AR module compared to the other standard classification methods.

ML algorithm	Decision Tree	Naive Bayes	Random Forest	EvAAL AR
F-measure	96.36%	97.93%	98.25%	99.04%
Precision	96.53%	97.75%	97.90%	98.85%
Recall	96.19%	98.11%	98.61%	99.22%

To obtain a better insight into which activities are misclassified, the confusion matrices for all the algorithms are presented in Fig 5.
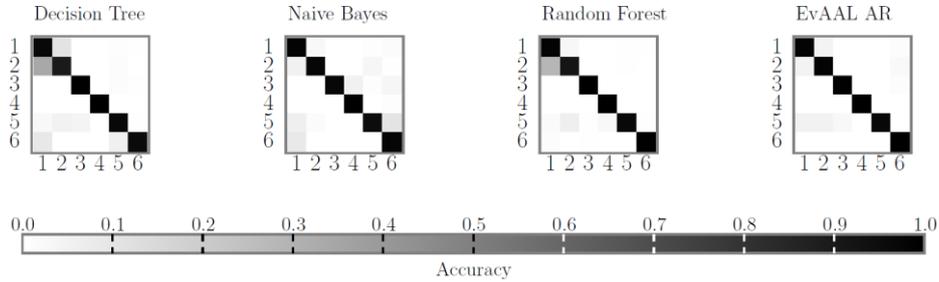
Fig. 5. Confusion matrices for all compared algorithms. Activity legend: 1 – walking, 2 – standing, 3 – sitting, 4 – lying, 5 – bending, 6– cycling.

Decision Tree and Random Forest classifiers have problems separating between walking and standing activities. This is probably due to the fact, that both activities have similar average values when computed over a time window. Naive Bayes classifier however fails at accurately recognizing cycling and walking. Although AR module developed for EvAAL has misclassified some standing and walking activities, it still outperforms other methods.

6.2 Fall Detection

The FD results (shown in Table 2) show that the first event – tripping (quick uncontrolled fall) is detected each time (15 out of all 15 events). The next event, i.e., fainting, is detected 13 out of 15 times. The next two events are the non-fall events that are difficult to distinguish from the fast falls because of the huge acceleration. Because the FD module also checks the user's orientation after a potential fall (huge acceleration), it was able to distinguish quickly sitting on the chair from the classical falls (only 1 false detection). However, the problem still exists in the event in which the user quickly lies into the bed (13 false detections). For correct recognition of this event, additional information about the user is needed, e.g., user's location.

Table 2. Fall detection results.

Events	Detected/All		Recall	Precision	F-measure
Tripping	15/15				
Fainting	13/15				
Quickly lying	13/15		**93.33%**	**66.67%**	**77.78%**
Quickly sitting	1/15				
Other	0				

The overall performance of the FD method shows that 93.3% of the fall events were detected (recall value), 66.7% of all the fall detections were actually falls (precision value), giving a final F-measure of 77.8%.

7 Conclusion

This paper presented an approach for AR and FD that emphasizes robustness and real-time performance: it combines human-understandable rules with classifiers trained with ML. The rules can classify postures (static activities) quite accurately, and they ensure that the behavior of the system is as predictable as possible and that nothing unforeseen occurs when it is deployed in practice. However, since constructing understandable and accurate rules for dynamic activities is difficult or even impossible, ML classifiers are used for these activities. On the other hand, the FD method first recognizes the high acceleration fall pattern and then checks the user's orientation. If the orientation corresponds to lying, a fall is detected; otherwise the recognized activity from the AR module is used as system's output.

The system was tested on a dataset containing a wide range of activities, two different types of falls and two events easily mistaken for falls. The F-measure of the AR was 99 %, which is very good for only two accelerometers and leave-one-person-out cross-validation. It can partly be attributed to our AR architecture and partly to the high quality of the dataset, since great care was taken to include only technically perfect data (no corrupted signals or incorrect labels). The F-measure of the FD was 78 %, which is not so good, but acceptable given the difficulty of the task. The main problem was mistaking lying down quickly for a fall, but could hardly be avoided without knowing that the event took place on a bed (impossible with accelerometers only) and without being able to observe the lying after a suspected fall for some time (impossible due to the emphasis on real-time performance).

We are currently working on the ergonomics aspect of our system. The two accelerometers are presently attached to the user with elastic straps, but we plan to sew them into clothing, if this can be done without significantly decreasing the accuracy due to less secure attachment. It appears that there is not much room for improvement of the AR accuracy, but the FD accuracy is less satisfactory. Therefore we plan to investigate approaches to improve this, both with additional sensors or by allowing more time to determine the context of a potential fall, and by improvements of the FD method only.

8 References

1. United Nations 2009, World population ageing, Report
2. Bourouis, A., Feham, M., Bouchachia, A.: A new architecture of a ubiquitous health monitoring system: a prototype of cloud mobile health monitoring system. The Computing Research Repository (2012)
3. Lustrek, M., Kaluza, B., Cvetkovic, B., Dovgan, E., Gjoreski, H., Mirchevska, V. Gams, M.: Confidence: ubiquitous care system to support independent living. DEMO at European Conference on Artificial Intelligence, pp. 1013-1014 (2012)
4. Gregory, D. A., Anind, K. D., Peter, J. B., Nigel, D., Mark, S., Pete, S.: Towards a better understanding of context and context-awareness. In: 1st International Symposium Handheld and Ubiquitous Computing, pp. 304-307 (1999)

5. Vyas, N., Farringdon, J., Andre, D. Stivoric, J. I.: Machine learning and sensor fusion for estimating continuous energy expenditure. In: Innovative Applications of Artificial Intelligence Conference, pp. 1613-1620 (2012)

6. Hall, M.J., Fingerhut, L., Heinen, M.: National Trend Data on Hospitalization of the Elderly for Injuries, 1979-2001. American Public Health Association (APHA) (2004)

7. Tinetti, M.E., Williams, C.S.: Falls, Injuries Due to Falls, and the Risk of Admission to a Nursing Home. The New England Journal of Medicine, vol. 337, pp. 1279–1284 (1997)

8. Wild, D., Nayak, U.S., Isaacs. B.: How dangerous are falls in old people at home?. British Medical Journal (Clinical Research Edition), vol. 282, no. 6260, 1982, pp. 266–268 (1981)

9. Gjoreski, H., Luštrek, M., Gams, M.: Accelerometer Placement for Posture Recognition and Fall Detection. In The 7th International Conference on Intelligent Environments, pp. 47–54 (2011)

10. PAMSys-The physical activity monitoring system, 2012. Available: http://www.biosensics.com/pamsys.html.

11. Sukthankar, G., Sycara, K.: A cost minimization approach to human behavior recognition. In: 4th International Conference on Autonomous Agents and Multi-Agents Systems, pp. 1067-1074 (2005)

12. Kwapisz, J. R., Weiss, G. M., Moore, S. A.: Activity recognition using cell phone accelerometers. ACM SIGKDD Explorer, vol. 12, pp. 74-82 (2010)

13. Wu, H., Lemaire, E. D., Baddour, N.: Activity change-of-state identification using a Blackberry smartphone. Journal of Medical and Biological Engineering, vol. 32, pp. 265-272 (2012)

14. Lai, C., Huang, Y. M., Park, J. H., Chao, H. C.: Adaptive body posture analysis for elderly-falling detection with multisensors. IEEE Intelligent Systems, vol. 25, pp. 2-11 (2010)

15. Gjoreski, H., Luštrek, M., Gams, M.: Context-Based Fall Detection using Inertial and Location Sensors. In: International Joint Conference on Ambient Intelligence, Lecture notes in computer science, pp. 1-16 (2012)

16. Ravi, N., Dandekar, N., Mysore, P., Littman, M. L.: Activity recognition from accelerometer data. In: 17th Conference on Innovative Applications of Artificial Intelligence, pp. 1541-1546 (2005)

17. Jantaraprim, P., Phukpattaranont, P., Limsakul, C., and Wongkittisuksa, B. 2009. Evaluation of Fall Detection for the Elderly on a Variety of Subject Groups. In: i-CREATe (2009)

18. Shan, S., and Yuan, T.: A Wearable Pre-Impact Fall Detector Using Feature Selection and Support Vector Machine. In: 10th IEEE International Conference on Signal Processing, pp. 1686–1689 (2010)

19. Nguyen, T.-T., Cho, M.-C., and Lee, T.-S.: Automatic Fall Detection Using Wearable Biomedical Signal Measurement Terminal. In: 31st Annual International Conference of the IEEE EMBS, pp. 5203–5206 (2009)

20. Zhang, T., Wang, J., Liu, P., and Hou, J.: Fall Detection by Wearable Sensor and One-Class SVM Algorithm. In: Lecture Notes in Control and Information Science, vol. 345, pp. 858–863 (2006)

21. Shimmer sensor platform, http://www.shimmer-research.com

22. Tapia, E. M.: Using machine learning for real-time activity recognition and estimation of energy expenditure. Ph.D. Thesis, Massachusetts Institute of Technology (2008)

23. Kozina, S., Gjoreski, H., Gams, M., Lustrek, M.: Three-layer Activity Recognition Combining Domain Knowledge and Meta-classification. Journal of Medical and Biological Engineering, doi: 10.5405/jmbe.1321

24. Weka application programming interface (API), 2012, http://weka.sourceforge.net/doc