Activity Recognition System Using AMEVA Method

Luis M. Soria Morillo[1], Luis González-Abril[2],
Miguel A. Álvarez de la Concepción[1], and Juan A. Ortega Ramírez[1]

[1] Computer Languages and Systems Dept., University of Seville, 41012 Seville, Spain
{lsoria,malvarez,jortega}@us.es
[2] Applied Economics I Dept., University of Seville, 41018 Seville, Spain
luisgon@us.es

Abstract. This article aims to develop a minimally intrusive system of care and monitoring. Furthermore, the goal is to get a cheap, comfortable and, especially, efficient system which controls the physical activity carried out by the user. For this purpose an innovative approach to physical activity recognition is presented, based on the use of discrete variables which employ data from accelerometer sensors. To this end, an innovative discretization and classification technique to make the recognition process in an efficient way and at low energy cost, is presented in this work based on the χ^2 distribution. Entire process is executed on the smartphone, by means of taking the system energy consumption into account, thereby increasing the battery lifetime and minimizing the device recharging frequency.

1 Introduction

Just 30 minutes of moderate activity five days a week, can improve your health according to the Centers for Disease Control and Prevention. By enabling activity monitoring on an individual scale, over an extended period of time in a ubiquitous way, physical and psychological health and fitness can be improved. Studies performed by certain health institutes initiative [7,3,10,6] have shown significant associations between physical activity and reduced risk of incident coronary heart disease and coronary events. Their results can be seen in Figure 1, where the inverse correlation between the risk of cardiovascular incidents and physical activity level is shown through the comparison of four separate studies.

In recent years, thanks largely to the increased interest in monitoring certain sectors of the population such those of as elderly people with dementia and of people in rehabilitation, activity recognition systems have increased in both number and quality. Furthermore, communication between relatives, friends and professionals can be improved by means of graphs of weekly activity (high relevant for sportsmen and for the relatives of elderly people) whereby the doctor can be automatically alerted if any strange activity is detected. By using data acquired from accelerometer, *NFC*, or even microphone sensors and applying

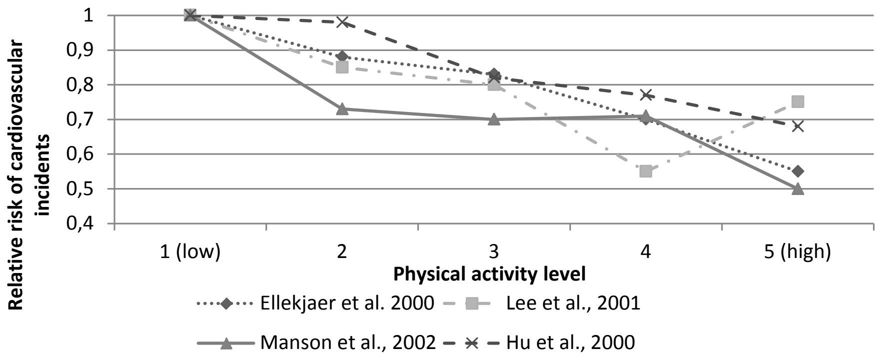

Fig. 1. Associations between physical activity and reduced risk of incident coronary heart disease and coronary events

some classification algorithm, it is possible to recognize human activities. Artificial neural networks (ANN) method will be analyzed and compared with our work. Results show the main differences between different studies, and certain drawbacks are determined which rules them out for development on users' smartphones To reduce the cost related to process accelerometer signals, this paper opts for an innovative technique, through which the work is performed in the field of discrete variables. Thanks to a discretization process, the classification cost is much lower than that obtained when working with continuous variables. Any dependence between variables during the recognition process is therefore eliminated and, on the other hand, energy consumption from the process itself is minimized.

2 Activity Recognition

2.1 Data Collection

Certain related studies attain results on activity recognition off-line. A comprehensive training set from the accelerometer output is first needed before data can be classified into any of the recognized activities. However, this paper has sought to minimize the waiting time for recognition, thereby providing valid information of the activity very frequently. To this end, both training and recognition sets are obtained using time windows [8] of fixed duration. After having conducted a performance and system accuracy analysis, it is determined that the optimum length for these windows is 5 seconds. Five seconds windows was chosen due to for our system it's extremely important to ensure that in each time window there is, at least, one activity cycle. Where activity cycle is define as an complete execution of some activity pattern. For instance, two steps are an activity cycle for walking and one pedal stroke is the activity cycle for cycling. If at least one activity cycle can not be ensure in each time window, it's not possible to determine, basing on accelerometer patterns, the activity performed. This statement could be seen in the next example. Suppose a two second cycle is having and the actor is jumping continuously, that is, we have a cadence of one jump for each two seconds. The system is configured with one second time window and thus, for each activity cycle will have two windows. In the first one, while the user is rising, vertical acceleration is negative. In the other one, because the user is falling, vertical acceleration will positive. If user increase the cadence by two, mean between acceleration set is close to , due to vertical positive and negative accelerations will be counteracted. For this reason, it's very important to ensure that one cycle of all activities, regardless of the speed performed, is contained in a time window. Segmentation process and activity cycle is shown in Figure 2.

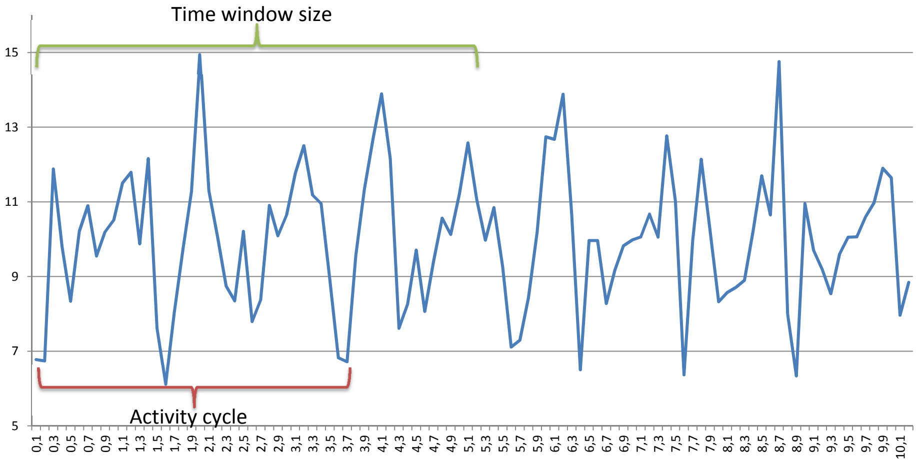

Fig. 2. Time windows split method over accelerometer signal

Based on these time windows, which contain data for each accelerometer axis, the signal module has been chosen in order to reduce the computational cost of the new solution. In addition to rendering the system more efficient, this choice of module eliminates the problem caused by device rotation [5,4]. Furthermore, user comfort with the system is decreased by removing the restriction that forces its orientation to be maintained during the process of learning and recognition. Using the accelerometer module, a data from each of the different readings taken within a time window $a_i = (a_{x,i}, a_{y,i}, a_{z,i})$ for the x, y, and z axes is defined as follows

$$|a_i| = \sqrt{(a_{x,i})^2, (a_{y,i})^2, (a_{z,i})^2} \tag{1}$$

For each temporal window is obtained Arithmetic Mean, Minimum, Maximum, Median, Std deviation, Geometric mean and other measures. In addition to the above variables, hereafter called temporal variables, a new set of statistics from the frequency domain of the problem is generated. This second set of variables will be called frecuencial variables. In order to obtain the frequency characteristics, the Fast Fourier Transform (*FFT*) for each time window is applied. In this way, and based on the frequency components obtained.

2.2 Set of Activities

Far from being a static system, the number and type of activities recognized by the system depends on the user [9]. However, to carry out a comparative analysis of the accuracy and performance of the discrete recognition method proposed below, 8 activities were taken into account. These activities are immobile, walking, running, jumping, cycling, drive, walking-upstairs and walking-downstairs. The learning system allows the user to decide what activities he/she wants the system to recognize. This is highly useful when the determination of certain very specific activities on monitored users is required. Examples of this situation include patients in rehabilitation who are monitored during their period of learning the various physical tasks prescribed by their doctors.

3 Qualitative Method

3.1 Ameva Algorithm

Let $X = \{x_1, x_2, \ldots, x_N\}$ be a data set of a continuous attribute $\mathcal{X}$ of mixed-mode data such that each example x_i belongs to only one of ℓ classes of the variable denoted by

$$\mathcal{C} = \{C_1, C_2, \ldots, C_\ell\}, \quad \ell \geq 2$$

A continuous attribute discretization is a function $\mathcal{D} : \mathcal{X} \to \mathcal{C}$ which assigns a class $C_i \in \mathcal{C}$ to each value $x \in \mathcal{X}$ in the domain of the property that is being discretized.

Let us consider a discretization $\mathcal{D}$ which discretizes the continuous domain of $\mathcal{X}$ into k discrete intervals:

$$\mathcal{L}(k; \mathcal{X}; \mathcal{C}) = \{[d_0, d_1], (d_1, d_2], \cdots, (d_{k-1}, d_k]\}$$

In this discretization, d_0 is the minimum value and d_k is the maximum value of the attribute $\mathcal{X}$, and the d_i values are in ascendent order.

If L_1 is the interval $[d_0, d_1]$ and L_j is the interval $(d_{j-1}, d_j]$, $j = 2, 3, \ldots, k$, then

$$\mathcal{L}(k; \mathcal{X}; \mathcal{C}) = \{L_1, L_2, \cdots, L_k\}$$

Therefore, the aim of the Ameva method [1] is to maximize the dependency relationship between the class labels $\mathcal{C}$ and the continuous-values attribute $\mathcal{L}(k)$, and at the same time to minimize the number of discrete intervals k.

As a result from applying the above algorithm to each statistical value of the system, a series of intervals associated with a particular $\mathcal{C}$ tag is obtained. Thus, after processing all system statistics, a three-dimensional matrix is obtained. In the first two dimensions, the label of the activity $\mathcal{C}$ associated with the interval

$L_i = (L_i^l, L_i^s]$, as well as with the lower limit L_i^l and the upper limit L_i^s of that range is stored. In a third dimension, the matrix contains the above data for each statistic $\mathcal{S} = \{S_1, S_2, ..., S_{\mathbb{S}}\}, \mathbb{S} \geq 2$. This three-dimensional matrix containing the set of interval limits for each statistic is called the *Discretization Matrix* and is denoted by $Dm\{\mathcal{C}, L^{l,s}, \mathcal{S}\}$. The *Discretization Matrix* therefore determines the interval to which each item of data belongs with respect to each statistical value, by means of carrying out a simple and fast discretization process.

Class Integration. The next step of the algorithm determines the probability associated with the statistical data for each of the activities based on previously generated intervals. To this end, each element of the training set $x = \{\mathcal{X}; \mathcal{C}\}$ is processed, to which, in addition to the value of each statistic whose calculation is based on the time window, is also associated the label of the specific activity in the training set. In order to carry out this process, *Class-Matrix* is denoted by $Cm\{x, L_i, \mathcal{S}\}$ and is defined as a three-dimensional matrix that contains the number of data x from the training set associated with each L_i interval for each statistical $\mathcal{S}$ of the system. This matrix is defined as follows,

$$Cm_{x,i,s} = |x \in \mathcal{X}| x \geq L_i^l \wedge x < L_i^s \wedge x\{\mathcal{C}\} = C_s \tag{2}$$

Therefore, by this definition, each position in the *Class-Matrix* is uniquely associated with a position in the *Discretization-Matrix*, as determined by its range.

At this point not only is it possible to determine the discretization interval likelihood, but the *Class-Matrix* also helps to obtain the probability associated with the discretization process performed with the *Ameva* algorithm.

Activity-Interval Matrix. The next step in the learning process is to obtain the matrix of relative probabilities. This three-dimensional matrix, called the *Activity-Interval Matrix* and denoted by $AIm\{x, L_i, \mathcal{S}\}$, determines the likelihood that a given value x associated to an $\mathcal{S}$ statistic corresponds to a specific C_i activity. This ratio is based on the quality of the discretization performed by *Ameva*, and in order to determine the most probable activity from the generated data and the intervals of the training set. First the contents of the array AIm is defined as follows,

$$AIm_{c,i,s} = \frac{Cm_{c,i,s}}{total_{c,s}} \cdot \frac{1}{\ell - 1} \sum_{j=1, j \neq c}^{\ell} (1 - \frac{Cm_{j,i,s}}{total_{j,s}}) \tag{3}$$

where $total_{c,s}$ is the total number of time windows of the training process labeled with the c activity for the $\int$ statistic.

Figure 3 shows the overall process described on this section for carry on data analysis and interval determination.

3.2 Classification Process

Having obtained the discretization intervals and the probabilities of belonging to each interval, the process by which the classification is performed can be

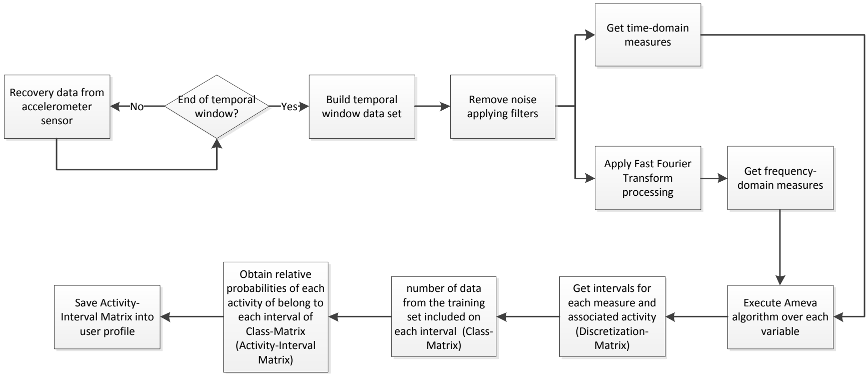

Fig. 3. Overall process of data analysis and interval determination

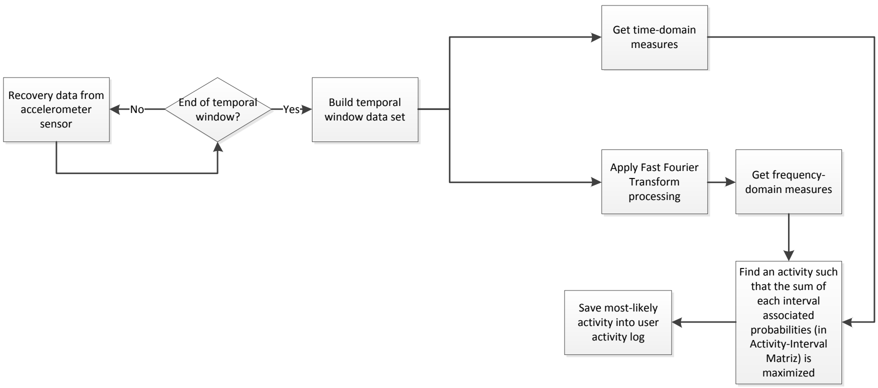

Fig. 4. Overall recognition process from data sensors

described. This classification is based on data from the analysis of time windows. The process is divided into two main steps: the way in which to perform the recognition of physical activity is first described; and the process to determine the frequency at which some particular activity is then presented.

Classifying Data. For the classification process, the most probable activity is decided by a majority voting system. This process starts from the *Activity-Interval Matrix* and uses a set of data $x \in \mathcal{X}$ for each of the statistics belonging to the $\mathcal{S}$ set. The process consists of finding an activity $mpa \in \mathcal{C}$ such that the likelihood is maximized. The above criterion is included in the following expression,

$$mpa(\mathcal{X}) = \max \sum_{s=1}^{s} AIm_{c,i,s} | x_s \in (L_i^l, L_i^s] \tag{4}$$

The expression shows that the weight contributed by each statistic to the calculation of the probability is identical. This can be carried out under the assumption that all statistics provide the same information to the system, and that there is no correlation between them. Thus, the most probable activity, or *mpa*, represents those activities whose data, obtained through the processing time window, is more suited to the *AIm* set values. In this way, the proposed algorithm not only determine the *mpa*, but also its associated probability. From this likelihood, certain activities that do not adapt well to sets of generic classification can be identified. This could be an indication that the user is carrying out new activities for which the system has not been previously trained.

Figure 4 shows the overall process described on this section for recognition process from Activity-Interval Matrix calculated in the previous stage.

4 Method Analysis

Now that the basis of the activity recognition algorithm has been laid out, an analysis of the new proposal can be performed. To this end, the new development is compared with a widely used recognition system based on neural networks [2]. In this case, both learning and recognition is performed by continuous methods. The test process is conducted on *Google Nexus S, Samsung Galaxy S2, and Google Nexus One* devices for a group of 40 users. Notably, the activity habits of these users are radically different, since 10 of them are under 25 years old, 20 users are between 25 and 40 years old, and the rest are over 40. An approximate distribution of the data for each subject regarding the eight activities in the study are: immobile (2800 min, 70 min per user), walking (2600 min, 65 min per user), running (2400 min, 60 min per user), jumping (2400 min, 60 min per user), cycling (2200 min, 55 min per user), driving (2200 min, 55 min per user), walking-upstairs (2400 min, 60 min per user), and walking-downstairs (2000 min, 50 min per user). Annotations are performed using a mobile application installed on the device itself with speech recognition software through which users dictate the name of the new activity when the physical activity being performance changes. Those unrecognizable activities conducted during the test process are dismissed to analyze the system accuracy. Data collection is obtained during four weeks.

Moreover, it is crucial to consider energy consumption and the processing cost of the system when it is working on a mobile device. In this case, after comparing the above methods, the conclusion reached is that the method based on *Ameva* reduces the computational cost of the system by about 50%, as can be seen in Figure 5. The time needed to process a time window by using the *Ameva*-based method is 0.6 seconds, while, for methods based on neural networks this figure is 1.2 seconds.

As can be seen in 6, Ameva battery consumption is lower than neural networks. For the first one, the battery lifetime is close to 25 hours while for the last one, it's only 16 hours. In the comparison can be observed the battery lifetime for decision tree but the main problem of this method, based on statistics chosen, is the low accuracy, not higher than 60%.

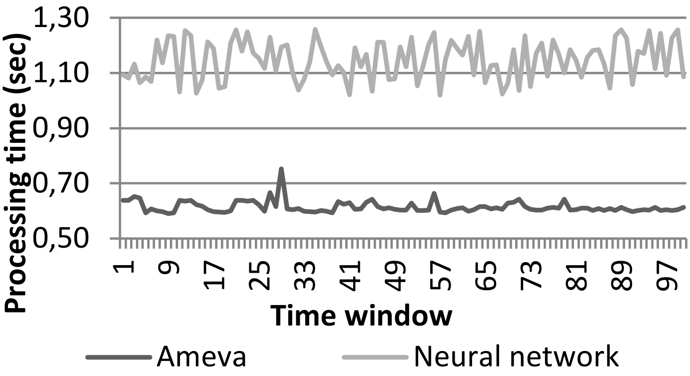

Fig. 5. Processing time of the Ameva and neural network methods on the device

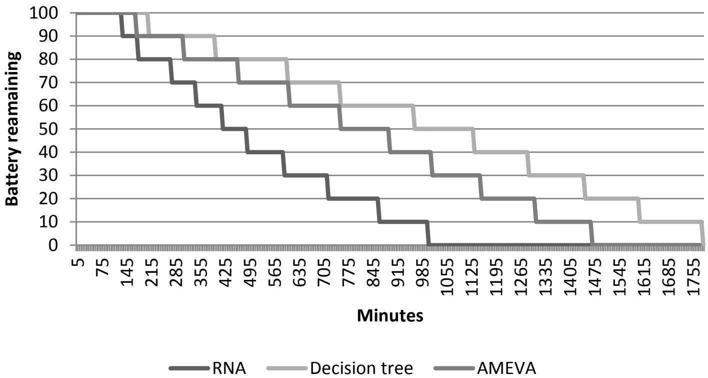

Fig. 6. Battery life for Ameva compared to neural network and decision tree methods

Based on Accuracy, Recall, Specificity, Precision, and F measure, Table 1 is presented. In this table, differences between the two methods, *RNA* and *Ameva* can be observed. Most values presented for each measure and activity show that the *Ameva* method performs better than *RNA*, especially as regards precision. That is to say, the number of false positive in the *Ameva* method is lower than that using the *RNA* method. *Immobile* and *Drive* are controversial activities due to their similar characteristics. Even under observation, it is difficult to differentiate between these two activities. For this reason and due to temporal nature of the *Immobile* activity, results from these two activities present a high level of disturbance in contrast to other activities.

Table 1. Performance comparison by using measures of evaluation

Activity	Accuracy		Recall		Specificity		Precision		F-measure (F_1)	
	Ameva	RNA	Ameva	RNA	Ameva	RNA	Ameva	RNA	Ameva	RNA
Walk	98.77%	97.93%	97.92%	93.95%	98.91%	98.57%	93.50%	91.36%	95.66%	92.64%
Jump	99.35%	98.87%	97.03%	96.44%	99.70%	99.25%	98.00%	95.12%	97.51%	95.77%
Immobile	98.69%	99.50%	94.57%	97.37%	99.42%	99.88%	96.60%	99.29%	95.58%	98.32%
Run	99.27%	98.35%	97.61%	92.62%	99.49%	99.14%	96.36%	93.64%	96.98%	93.13%
Up	98.93%	98.17%	95.40%	90.79%	99.43%	99.22%	96.00%	94.35%	95.70%	92.54%
Down	98.64%	98.25%	95.20%	92.68%	99.04%	98.89%	91.95%	90.62%	93.55%	91.64%
Cycle	99.32%	99.03%	96.13%	95.67%	99.73%	99.47%	97.91%	95.89%	97.01%	95.78%
Drive	98.14%	98.74%	90.02%	95.01%	99.20%	99.23%	93.63%	94.16%	91.79%	94.58%

5 Conclusions and Future Work

In this work, a highly successful recognition system based on discrete variables is presented, which uses the *Ameva* discretization algorithm and a new *Ameva*-based classification system. It has therefore been possible to achieve an average accuracy of 98% for the recognition of 8 types of activities. Furthermore, working with discrete variables has significantly reduced the computational cost associated to data processing during the recognition process. By using this process to increase recognition frequency, it has been possible to obtain a physical activity reading every 5 seconds and to enter these readings into the user activity log. However, the main problem of this system based on statistical learning is the limit to the number of activities that can be recognized. Working only with accelerometer sensors implies a limit to the number of system variables and therefore may lead to a strong correlation between these variables.

6 AMEVA Running in EvAAL Competition

During the competition, two test sessions were executed. In the first one, the training was performed prior to competition by an external actor not related to evaluation process. The training actor was 31 years old and the entire training process was performed with the smartphone in the hip, attached to the user's belt. In the competition, the actor was in a similar age range and thus, the way in that physical activity was executed was very similar. In other case, the system should be retrained for a better accuracy. Once finished the first evaluation session, intermediate data was analyzed. From this analysis, it was concluded that some activities was not well-recognized such as bending or cycling. This was a substantial impact in the accuracy due to cycling session was long. The accuracy for the other activities was promising but we detect that something was wrong for cycling detection. By using discrete techniques to perform the activity recognition, cycling is a easy activity to be detected because of the acceleration patterns presents an evident component in the advancing direction. Unfortunately, cycling activity was carried out on a stationary bike and thus, accelerations presented in movement direction was not detected. For the other controversial activity, bending, the system was not training to detect it because it was a important conflict with sitting activity. Both activities have a very similar acceleration profile and it can not be determine which is the right activity with a proper accurate. In the second test process, the system was retrained in order to achieve a most accurate recognition. Unfortunately, the Internet connection was not good enough to connect with training server placed at the University of Seville. For this reason, dataset from time windows was not properly sent to the server and therefore, the training parameters were wrong. After checking this problem, we decided to go on with the evaluation process to determine the impact of this problem in the accuracy. As it was thought, the second evaluation had a very low accuracy due to that problems. Furthermore, by studying intermediate data after the evaluation, temporal windows was misconfigured and it was set to 3 seconds and thus, some "fast activities" such as walking or cycling wasn't well recognized. Finally, EvAAL competition was a great chance to make a real stress test of AMEVA system since It's not usual in humans to make a long activities set in so quickly and so fast. In this regard, statistical-discrete classification for activity recognition based on AMEVA algorithm was designed to medium-long time activities. Transitions in discrete classification systems are really difficult to detect and, in AMEVA case, was not implemented any change activity detector. In conclusion, EvAAL offered a junction to test many systems and generate new ideas for competitors' systems. On the other hand, is very good to know other techniques in activity recognition and new perspectives about this field.

Acknowledgments. This research is partially supported by the projects of the Spanish Ministry of Economy and Competitiveness ARTEMISA (TIN2009-14378-C02-01) and Simon (TIC-8052) of the Andalusian Regional Ministry of Economy.

References

1. González Abril, L., Cuberos, F.J., Velasco, F., Ortega, J.A.: Ameva: An autonomous discretization algorithm. Expert Syst. Appl. 36(3), 5327–5332 (2009)
2. Altun, K., Barshan, B., Tunçel, O.: Comparative study on classifying human activities with miniature inertial and magnetic sensors. Pattern Recogn. 43(10), 3605–3620 (2010)
3. Ellekjaer, Holmen, Vatten: Physical activity and stroke mortality in women. Stroke 31(1), 14–18 (2000)
4. He, Z., Jin, L.: Activity recognition from acceleration data based on discrete consine transform and SVM. In: 2009 IEEE International Conference on Systems, Man and Cybernetics, pp. 5041–5044. IEEE (October 2009)
5. Khan, A.M., Lee, Y.-K., Lee, S., Kim, T.-S.: Accelerometer's position independent physical activity recognition system for long-term activity monitoring in the elderly. Medical & Biological Engineering & Computing 48(12), 1271–1279 (2010)
6. Lee, I.-M., Rexrode, K.M., Cook, N.R.: Physical activity and coronary heart disease in women. JAMA 285(11), 1447–1454 (2001)

7. Manson, J.E., Greenland, P., LaCroix, A.Z., Stefanick, M.L., Mouton, C.P., Oberman, A., Perri, M.G., Sheps, D.S., Pettinger, M.B., Siscovick, D.S.: Walking compared with vigorous exercise for the prevention of cardiovascular events in women. The New England Journal of Medicine 347(10), 716–725 (2002)
8. Paiyarom, S., Tungamchit, P., Keinprasit, R., Kayasith, P.: Activity monitoring system using Dynamic Time Warping for the elderly and disabled people, pp. 1–4. IEEE (2009)
9. Ravi, N., Nikhil, D., Mysore, P., Littman, M.L.: Activity recognition from accelerometer data. In: Proceedings of the Seventeenth Conference on Innovative Applications of Artificial Intelligence(IAAI), pp. 1541–1546 (2005)
10. Sattelmair, J.R., Kurth, T., Buring, J.E., Lee, I.-M.: Physical Activity and Risk of Stroke in Women. Stroke 41(6), 1243–1250 (2010)