

Discrete techniques applied to low-energy mobile human activity recognition. A new approach

Álvarez de la Concepción M.A., Soria Morillo L.M. , Gonzalez-Abril L. , Ortega Ramírez J.A.

A B S T R A C T

Keywords:

Pattern recognition
Discretization method
Qualitative systems
Smart-energy computing
Energy saving

Human activity recognition systems are currently implemented by hundreds of applications and, in recent years, several technology manufacturers have introduced new wearable devices for this purpose. Battery consumption constitutes a critical point in these systems since most are provided with a rechargeable battery. In this paper, by using discrete techniques based on the Ameva algorithm, an innovative approach for human activity recognition systems on mobile devices is presented. Furthermore, unlike other systems in current use, this proposal enables recognition of high granularity activities by using accelerometer sensors. Hence, the accuracy of activity recognition systems can be increased without sacrificing efficiency. A comparative is carried out between the proposed approach and an approach based on the well-known neural networks.

1. Introduction

In recent years, thanks largely to the growing interest in monitoring certain sectors of the population, such as elderly people with dementia and people in rehabilitation, activity recognition systems have experienced an increase in both number and quality of results. However, most of these results incur high computational costs, and hence cannot be applied on a general-purpose mobile device due to their excessive energy consumption.

Although, the calculation of the physical activity of a user, based on data obtained from an accelerometer, remains a current research topic, numerous limitations have been identified that make these systems uncomfortable for users in general.

The first difference observed between the many systems developed is the type of sensor used. There are systems using specific hardware (Ravi, Dandekar, Mysore, & Littman, 2005), while others use general-purpose hardware (Hong, Kim, Ahn, & Kim, 2008). Obviously, the use of generic hardware constitutes a benefit for users, since their availability, low cost, and versatility are points greatly in their favour; not to mention the reduction in the risk of loss, since these devices have already been integrated into everyday objects, such as users' smartphones. However, general purpose devices are used for other purposes, such as making phone

calls, surfing the Internet, and listening to music. For this reason, the physical activity recognition system must be executed in background mode and cause the least impact on the system as possible, in terms of complexity and energy consumption.

Another difference found between the proposals surveyed is the number and position of the sensors. In Brezmes, Gorricho, and Cotrina (2009), it can be observed that the accelerometer sensor is placed in a glove and a multitude of activities are recognized depending on the movement of the hand. In contrast, other studies use either various sensors placed on many parts of the body (Bicocchi, Mamei, & Zambonelli, 2010; Lepri, Mana, Cappelletti, Pianesi, & Zancanaro, 2010) or a wearable wireless sensor node with a static wireless non-intrusive sensory infrastructure (Paoli, Fernández-Luque, & Zapata, 2011) to recognize these activities. According to certain comparative studies and previous research based on multiple sensors, these last two types of sensors provide greater accuracy.

However, in studies, such as Hong et al. (2008), a sensor is kept in a user's pocket or worn on the hip, which is more wearable on the monitored person, and requires much lower infrastructure.

Once the most comfortable alternative for users is determined, some device sensors could be chosen to perform the activity monitoring. Some research uses data not only from accelerometers and a gyroscope (Dernbach, Das, Krishnan, Thomas, & Cook, 2012), but also from other sources, such as microphones, light sensors and voice recognition to determine the context of the user (Kwapisz, Weiss, & Moore, 2011) and ECG sensors (Li et al., 2010; Pawar, Chaudhuri, & Dutttagupta, 2007; Ward, Lukowicz, Troster, & Starner, 2006) for this purpose.

The first example, which incorporates microphone and blue-tooth devices, helps to obtain contextual information about the user environments and would be appropriate to perform a more in-depth analysis of the activity, for instance, whether the user is walking in a disco or at home, or whether s/he is alone or with someone. However, high-level activity recognition (walking, playing, running or standing up) is carried out using other sensors. On the other hand, ECG can help determine high-level activities by means of heart-rate processing. In this sense, certain activities (walking or running) could be discerned based on the effort exerted. However, the problem here is that ECG sensors are both expensive and uncomfortable for the user.

Thus, the present work is focused on the recognition of physical activities carried out by users by means of their mobile devices, and hence special attention must be paid to energy consumption and the computational cost of the methods used.

There are related studies where data for activity recognition is obtained through mobile devices where this data is sent to a server to process the information (Altun, Barshan, & Tunçel, 2010). In these cases, computational cost is no limitation and hence methods of a more complex nature can be used. In contrast, efficiency is a crucial issue when processing is carried out within the mobile device itself (Fuentes, Gonzalez-Abril, Angulo, & Ortega, 2012; Reddy et al., 2010).

Taking previous works into account, physical activity monitoring through smartphones presents the following challenges:

- To decrease, as far as possible, the risk of forgetting the processing device so that continuous monitoring can be performed anywhere and any time.
- To reduce the drain of energy on the smartphone, by developing an accurate and efficient system.
- To integrate learning and monitoring on the device itself, in realtime and without sharing server information.

In order to reduce the cost associated to accelerometer and gyroscope signal analysis, this paper opts for an innovative approach based on a discretization method that uses only accelerometer sensors since with these good results can be achieved and a lot of energy can be saved (the more sensors used, the greater the consumption). Furthermore, the data is processed in the mobile itself, and therefore the efficiency is better than if data were sent to a server, and the result is obtained in real time. Thanks to this discretization process, the classification cost is much lower than it would be with continuous variables, and therefore the life of the battery is longer. It is therefore possible not only to eliminate the correlation between variables during the recognition process, but also to minimize the energy consumption of the process.

The remainder of the paper is organized as follows: Section 2 describes the method of data capture through the mobile device using an accelerometer sensor, and outlines all the physical activities that can be recognized by the system described. In Section 3, a new process, that discretizes continuous variables and provides classification, is presented. Section 4 shows a comparison between the new method and other methods used previously in the literature. Finally, in Section 5, a discussion is given on the various advantages of the proposed algorithm as well as on certain challenges and tasks that are currently being developed for the described recognition system.

2. Activity recognition

2.1. Embedded sensors and battery impact

Throughout this work, a KR3DM triaxial accelerometer integrated into a Google Nexus S is used. This sensor has a range of

sampling frequencies between 25 Hz and 1500 Hz. Specific frequency is defined by each piece of software by using Android primitives. At these sampling rates, the device used for the testing process is configured to operate at 50 Hz in order to prevent excessive use of data and to reduce the computational cost. Therefore, based on the Nyquist–Shannon theorem, it can be ensured that signals with significant energy components below 25 Hz are liaison free. Depending on where the user takes the device, vibrations and any other kind of noise could be present with frequency components over 25 Hz, but these are of no interest in this work. However, the human-activity frequency range is much lower than the sampling band chosen. By using accelerometers taped to the body while running, Bhattacharya, McCutcheon, Shvartz, and Greenleaf (1980) found the main frequency components between 1–18 Hz at the ankle. As will be seen later, our work proposes placing the device at the hip, where acceleration forces are lower than at the ankle, and hence frequencies at this position are also lower.

On the other hand, a lower frequency allows the computation cost to be reduced thanks to the fact that each feature is obtained from accelerometry data. Furthermore, by reducing this processing time, the system becomes faster, more efficient and consumes less energy. Indeed, for contextual systems, with their intensive use of sensors, the high-energy consumption required must be taken into account not only in obtaining the data but also in its processing. A typical smartphone from the latest generation has a multitude of sensors that are commonly used, such as GPS (Morillo, Ramirez, Garcia, & Gonzalez-Abril, 2012), NFC, and a microphone. This means that, as result of high energy consumption, the useful time between device charges remains very low.

By applying Moore's law, it can be observed that manufacturers increase their processing power at least twice each year, in contrast to battery development, which has failed to double over the last five years. Battery life is not a secondary consideration, since, according to a survey performed by North American Technologies (Forrester, 2011), it stands as the second most important purchase decision factor for buyers of smartphones. Users acceptance, in the context of aware applications in general and of activity recognition systems in particular, is therefore critical. For this reason, in this work, not only has an accurate and fast system been developed, but a low energy consumption model is also presented from the viewpoint of discrete techniques.

There are various solutions using specific hardware (Choudhury et al., 2008) that have a high degree of autonomy. However, the problems faced by these elements are, as outlined earlier, the risk of losing and/or forgetting the device and the discomfort for users. Furthermore, these solutions are usually very expensive. In recent years, a large number of personal devices able to monitor the activity level have been developed by large companies, such as Adidas (Fig. 1), and Nike (Fig. 2). However, the aims of this work, even though related, are quite different. Both of these commercial devices allow the physical activity level to be detected. This is interesting from the point of view of calories burned. The limit imposed on the number of activities constitutes the main disadvantage of these systems, since they can only detect certain parameters, such as the number of sprints or total distance covered. Although these features are significant, for many users the number of these activities may be insufficient, for example, for users involved in a physical rehabilitation process which must be monitored by doctors. In these cases, activity recognition should have greater granularity to detect activities. The activities our system is able to discern will be discussed in the following section.

In short, although our proposal has been tested with smartphone embedded sensors, no limitation whatsoever is envisaged. In this paper, a new independent-sensor technique for activity recognition is presented, which uses and continues the work in Soria Morillo, Ortega Ramirez, and Gonzalez-Abril (2012). Thus, this



Fig. 1. Adidas MyCoach: intalled into Adidas shoes.



Fig. 2. Nike FuelBand: wristband with accelerometer sensor for activity level.

technique can be developed on a smartphone, a commercial wristband or any other appliance which contains an accelerometer and processing unit. In contrast to other proposals, our work is focused on battery consumption, which becomes even more critical on external devices where drains on the battery must be reduced to provide greater usage time.

There are several related studies which centre on the accuracy power trade off, for example in the proposals by Zappi et al. (2008) and Wang et al. (2009). Not only do these proposals operate with smartphones, but also with distributed body sensors, where it is possible to determine which sensors are the most important, and which can be disconnected in order to minimize energy cost.

2.2. Set of activities

In this work, far from being a static system, the number and type of activities recognized depends on the user; users can carry out hot-training on the system, which may involve adding new activities, and hence other activities can be detected. This is crucial for the accomplishment of a highly customizable environment where the users themselves can determine which activities are important and which remain irrelevant.

For a large numbers of users, it could prove more practical to recognize only a few activities, such as walking, sitting, and falling. For other users, however, activities, such as driving and cycling could be more relevant. In a rehabilitation process, by setting out an exercise programme, doctors and relatives could ascertain whether the user is carrying out the prescribed physical exercise correctly.

During the system testing and by performing comparative analysis with other platforms, 8 activities have been taken into account. These activities include standing, walking, running, jumping, cycling, driving, climbing stairs, and descending stairs. Moreover, thanks to this proposal, activities that have not been previously learned by the system can be determined while users carry them out. This learning process is achieved based on the analysis of the probability associated to each pattern while the user is performing each activity. In this sense, the system alerts the user when a non-trained activity is detected and therefore, the opportunity to carry out in-depth training for this activity is presented.

Obviously, the number of activities to be detected has an impact on the accuracy of the system, especially if acceleration patterns between activities are very similar. This aspect will be discussed later.

2.3. Data collection

Several related studies attain results of activity recognition off-line. First, an appropriate set of acceleration readings must be obtained from sensors. Once this set is completed, it must be sent to the server and the readings are then classified into any of the recognized activities. In Duong, Bui, Phung, and Venkatesh (2005) and other model-driven studies into activity recognition, learning is conducted off-line. This means that the user must be connected to the training server so that the model, from which the recognition process can be carried out, can be obtained.

However, training and recognition sets are obtained using time windows of fixed duration. Each time window is composed of a set of accelerometer readings from which it is possible to calculate a variety of features. After having conducted a performance and system accuracy analysis, it has been determined that the optimum length for these windows is 5 s. This time has been chosen due to the importance of ensuring that, in each time window, there is at least one activity cycle. An activity cycle is defined as an complete execution of an activity pattern. For example, two steps are an activity cycle for walking and one pedal stroke is the activity cycle for cycling. If a complete activity cycle is not presented in each time window, then, based on acceleration patterns, it will not be possible to determine the activity performed. The segmentation process and activity cycle are shown in Fig. 3.

Based on these time windows, which contain data for each accelerometer axis and reduce the computational cost of the new solution, a signal module has been chosen. This eliminates the problem caused by the device rotation (He & Jin, 2009). Furthermore, it increases user comfort by removing the restriction of maintaining the same orientation during the learning and recognition process.

For each data in a time window size N , $a_i = (a_i^x, a_i^y, a_i^z)$, $i = 1, 2, \dots, N$ where x, y and z represent the three accelerometer axes, the accelerometer module is defined as follows:

$$|a_i| = \sqrt{(a_i^x)^2 + (a_i^y)^2 + (a_i^z)^2}$$

Hence, the following statistics are obtained for each time window.

- Mean: $\bar{a} = \frac{1}{N} \sum_{i=1}^N |a_i|$.
- Minimum: $a_{min} = \min\{|a_1|, |a_2|, \dots, |a_N|\}$.

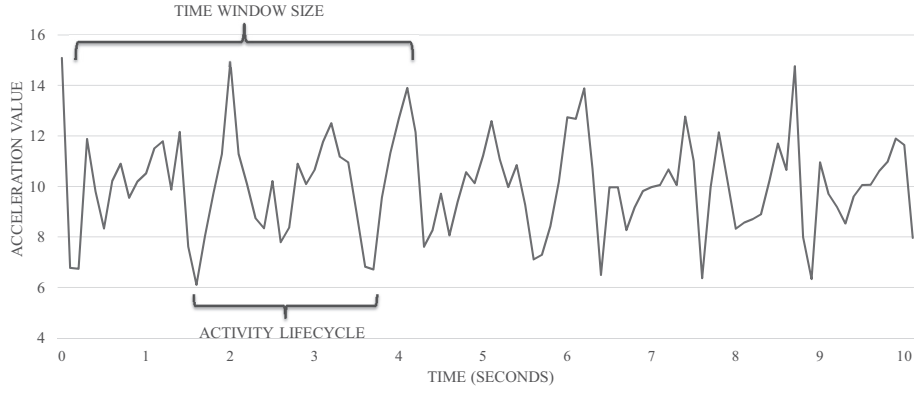


Fig. 3. Time windows split method over accelerometer signal.

- Maximum: $a_{max} = \max\{|a_1|, |a_2|, \dots, |a_N|\}$.
- Median: $Me = |a_{(N+1)/2}|$ if $N\%2 \neq 0$ and $\frac{|a_{N/2}| + |a_{(N+1)/2}|}{2}$ otherwise.
- Standard deviation: $S = \sqrt{\frac{1}{N} \sum_{i=1}^N (|a_i| - \bar{a})^2}$.
- Signal magnitude area (Mathie et al., 2004): $SMA = \sum_{i=1}^N (|a_i^x| + |a_i^y| + |a_i^z|)$.
- Mean deviation: $D_m = \frac{1}{N} \sum_{i=1}^N ||a_i| - \bar{a}|$.

In addition to the above variables, hereinafter called temporary variables, a new set of statistics called frequency-domain features are generated from the frequency domain of the problem. In order to obtain these frequency-domain features, the Fast Fourier Transform (FFT) is applied for each time window. Frequency components associated to each time window are defined as follows:

$$y_k = \sum_{j=0}^{F-1} |a_j| e^{-2\pi i j k}$$

where F is the cardinality of frequency components and $k = 1, 2, \dots, F$.

From here, the following attributes are identified:

- Min module: $m_{min} = \min\{|y_1|, |y_2|, \dots, |y_F|\}$.
- Max module: $m_{max} = \max\{|y_1|, |y_2|, \dots, |y_F|\}$.
- Min frequency module: $f_{min} = \{f(y_k) | dle | y_k = m_{min}\}$.
- Max frequency module: $f_{max} = \{f(y_k) | dle | y_k = m_{max}\}$.

where $|y_k|$ and $f(y_k)$ are the module and frequency associated to the k th component, respectively.

It cannot be forgotten that the entire process is to be executed on the smartphone itself. Accordingly, the number of features must be reduced to minimize the impact of the processing energy used. To reduce the number of features set out above, the PCA method has been applied. In this way, a final set of 48 features was selected. It should be borne in mind that the first ten components deliver more than 98% of the variance in all validations. Therefore, in order to reduce the complexity, henceforth just 10 components will be used to describe the system.

3. Methodology

To the best of our knowledge, our proposal, which involves working in the domain of discrete variables to perform learning and recognition of activities, constitutes a totally innovative approach. This concept arose largely due to the need for a reduction in the high computational cost required for learning algorithms based on continuous variables, which have been used for this purpose over the years.

In Gonzalez-Abril, Velasco, Ortega, and Cuberos (2009), a labeling process, similar to a discretization process, is used to obtain a Qualitative Similarity Index (QSI), and hence it can be said that a transformation of the continuous domain to the discrete domain of values of the variables is beneficial in certain aspects. Therefore, the Ameva discretization (Gonzalez-Abril, Cuberos, Velasco, & Ortega, 2009) is used, which is unsupervised and fast. The most notable feature of these two processes is the small number of intervals generated, which facilitates and reduces the computational cost of the recognition process.

Let us briefly examine these algorithms.

3.1. Ameva algorithm

Let $X = \{x_1, x_2, \dots, x_n\}$ be a data set of an attribute \mathcal{X} of mixed-mode data such that each example x_i belongs to only one of the ℓ classes whose class variable is denoted by

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

A continuous attribute discretization is a function $\mathcal{D} : \mathcal{X} \rightarrow \mathcal{C}$ which assigns a class $C_i \in \mathcal{C}$ to each value $x \in X$ in the domain of the property that is being discretized. Let us consider a discretization \mathcal{D} which discretizes \mathcal{X} into k discrete intervals:

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

where L_1 is the interval $[d_0, d_1]$ and L_j is the interval $(d_{j-1}, d_j], j = 2, 3, \dots, k$. Thus, a discretization variable is defined as $\mathcal{L}(k) = \mathcal{L}(k; \mathcal{X}; \mathcal{C})$ which verifies that, for all $x_i \in X$, a unique L_j exists such that $x_i \in L_j$ for $i = 1, 2, \dots, n$ and $j = 1, 2, \dots, k$. The discretization variable $\mathcal{L}(k)$ of \mathcal{X} and the class variable \mathcal{C} are treated from a descriptive point of view.

The main aim of the Ameva method (Gonzalez-Abril et al., 2009) 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 . To this end, the following statistic is used:

$$\text{Ameva}(k) = \frac{\chi^2(k)}{k(\ell-1)} \text{ where } \chi^2(k) = N - 1 + \sum_{i=1}^{\ell} \sum_{j=1}^k \frac{n_{ij}^2}{n_i n_j}$$

Here, n_{ij} denotes the total number of continuous values belonging to the C_i class that are within the interval L_j ; n_i is the total number of instances belonging to the class C_i ; and n_j is the total number of instances that belong to the interval L_j , for $i = 1, 2, \dots, \ell$ and $j = 1, 2, \dots, k$, which fulfill the following:

$$n_i = \sum_{j=1}^k n_{ij}, \quad n_j = \sum_{i=1}^{\ell} n_{ij}, \quad N = \sum_{i=1}^{\ell} \sum_{j=1}^k n_{ij}$$

3.2. Discretization process

For each statistic $S_p \in \{S_1, S_2, \dots, S_m\}$, the discretization process is performed, and a matrix of order $k_p \times 2$ is obtained, where k_p is the number of class intervals and 2 denotes the $\inf(L_i^p)$ and $\sup(L_i^p)$ interval limits i of the statistic p . Hence, a three-dimensional matrix containing the statistics and the set of interval limits for each statistic is called the Discretization Matrix and is denoted by

$$\mathcal{W} = (w_{pij})$$

where $p = 1, 2, \dots, m, i = 1, 2, \dots, k_p$, and $j = 1, 2$ (see Fig. 4).

Therefore, the Discretization Matrix determines the interval in which each item of data belongs for the different statistical associated values, by carrying out a simple and fast discretization process.

3.2.1. Class integration

The aim in the next step of the algorithm is to provide a probability associated with the statistical data for each of the activities, based on previously generated intervals. For this purpose, the elements of the training set $x \in X$ are processed to associate the label of the specific activity in the training set. In addition, the value of each statistic is calculated based on the time window.

In order to perform this process, a Class Matrix, \mathcal{V} , is defined as a three-dimensional matrix that contains the number of item of data from the training set associated with an \mathcal{L} interval in a \mathcal{C} activity for each statistic \mathcal{S} of the system. This matrix is defined as follows:

$$\mathcal{V} = (v_{pij})$$

where $v_{pij} = \#\{x \in X | \inf(L_i^p) < x \leq \sup(L_i^p)\}$, $S = S_p, C = C_j, p = 1, 2, \dots, m, i = 1, 2, \dots, k_p$ and $j = 1, 2, \dots, \ell$. Hence, each position in the Class Matrix is uniquely associated with a position in the Discretization Matrix, which is determined by its range.

Fig. 5 shows the contents of a real Class Matrix obtained during a learning process from all statistics. In order to simplify the figure for this example, only 3 out of 8 activities recognized by the system have been taken into consideration. Within these activities, five intervals determined by the Ameva algorithm can be observed in the Mean Class Matrix.

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

3.2.2. Activity-Interval Matrix

In the next step, a three-dimensional matrix, called the Activity-Interval Matrix is defined, denoted by \mathcal{U} , which specifies the likelihood that a given value x associated to a statistic S corresponds to

Fig. 5. Class matrices.

an activity \mathcal{C} in an interval \mathcal{L} . This ratio is based on the goodness of the Ameva discretization in order to determine the most probable activity from the data and the intervals generated for the training set.

Each value of \mathcal{U} is defined as follows:

$$u_{pij} = \frac{v_{pij}}{v_{p,j}} \frac{\sum_{q=1, q \neq j}^{\ell} (1 - \frac{v_{piq}}{v_{p,q}})}{\ell - 1}$$

where $v_{p,j}$ is the total number of time windows of the training process labelled with the j activity for the p statistic, and $p = 1, 2, \dots, m, i = 1, 2, \dots, k_p$, and $j = 1, 2, \dots, \ell$.

Given these values, \mathcal{U} for the p statistic is defined as

$$\mathcal{U}_p = \begin{pmatrix} u_{p00} & \dots & u_{p0j} & \dots & u_{p0\ell} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ u_{pi0} & \dots & u_{pij} & \dots & u_{pi\ell} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ u_{pk_p,0} & \dots & u_{pk_p,j} & \dots & u_{pk_p,\ell} \end{pmatrix}$$

As can be seen in the definition of \mathcal{U} , the likelihood that data x is associated with the interval L_i corresponding to the activity C_j , depends not only on the data, but on all the elements associated with the interval L_i for the other activities.

Thus, each u_{pij} matrix position can be seen as a degree of belonging for a given x , identified with a C_j activity, to be included in the L_i interval of the S_p statistic.

Similarly, the elements of \mathcal{U} hold the following properties:

- $u_{pij} = 0 \iff v_{pij} = 0 \vee v_{piq} = v_{p,q}, q \neq j$

Fig. 4. Discretization matrices.

Fig. 6. Activity-Interval matrices.

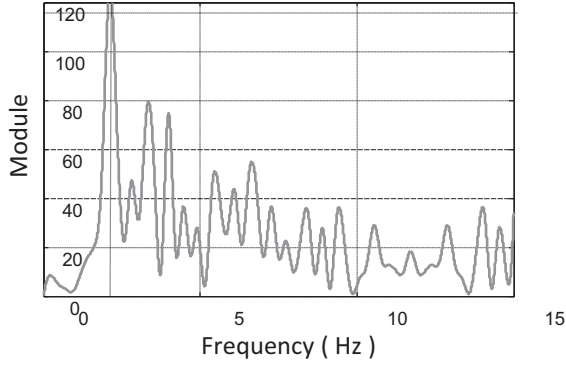


Fig. 7. Frequential study from accelerometer signal on a smartphone.

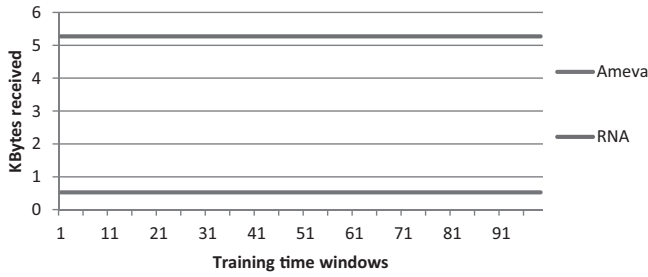


Fig. 8. Information flow between smartphone and server.

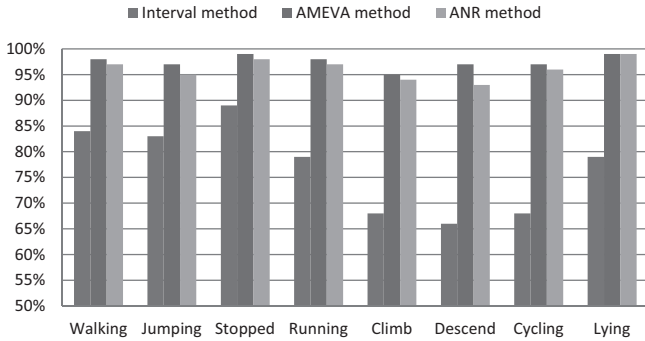


Fig. 9. Processing time of Ameva and neural network methods on the device.

- $u_{p_{ij}} = 1 \iff v_{p_{ij}} = v_{p_j} = v_{p_i}$.

Fig. 6 shows a set of values for each of the positions of the Activity-Interval Matrix. These results have been obtained from the training set of the Class Matrix described above in Fig. 5.

Table 1
Confusion Matrix.

Actual class	Predicted class																	
	Walk		Jump		Stop		Run		Climb		Descend		Cycle		Drive		Total	
	Ameva	RNA	Ameva	RNA	Ameva	RNA	Ameva	RNA	Ameva	RNA	Ameva	RNA	Ameva	RNA	Ameva	RNA	Ameva	RNA
Walk	1036	994	6	6	6	18	6	22	8	16	4	10	8	12	34	10	1108	1088
Jump	2	10	980	974	0	0	4	12	2	8	8	8	2	6	2	6	1000	1024
Immobile	0	0	0	0	1080	1112	0	0	0	0	6	0	8	0	24	8	1118	1120
Run	4	14	12	12	0	0	900	854	8	12	2	8	4	6	4	6	934	912
Climb	2	6	2	6	10	0	2	6	912	868	14	20	6	8	2	6	950	920
Descend	8	20	6	6	10	0	6	8	18	34	754	734	2	2	16	6	820	810
Cycle	2	6	2	4	0	0	2	14	4	6	2	4	844	840	6	2	862	876
Drive	4	8	2	2	36	12	2	6	4	12	2	8	4	4	794	838	848	890
Total	1058	1058	1010	1010	1142	1142	922	922	956	956	792	792	878	878	882	882	7640	7640

3.3. Classification process

After obtaining the discretization intervals and the various probabilities of belonging, this section describes the process of classification from the data of the analysis time windows. This process is divided into two main parts. First, the way to perform the recognition of physical activity is described, and then the task of determining the frequency of a particular activity is presented.

3.3.1. Classifying data

For the classification process, the most likely activity is decided by a majority voting system. As described above, this process starts from the Activity-Interval Matrix and a set of data $x \in X$ for the set S .

The process therefore consists of finding an activity $C_i \in C$ that maximizes this likelihood. This criterion is collected in the following expression, denoted by mpa (most probable activity):

$$mpa(x) = C_k$$

where $k = \arg(\max_j \sum_{p=1}^m u_{p_{ij}} | x \in (\inf(L_i^p), \sup(L_i^p)))$. The expression shows that the weight, contributed by each statistic to the probability calculation function, is the same. This expression assumes that all statistics provide the same information to the system and there is no correlation between them.

Thus, the mpa represents the activity whose data, obtained within the processing time window, is more suited to the set of values from \mathcal{U} . In this way, the proposed algorithm not only determines the mpa , but also determines its associated probability.

From this likelihood, certain activities that fail to adapt well to sets of generic classification can be identified, and constitute an indication that the user is carrying out new activities for which the system has not been previously trained.

3.3.2. Frequency activity approach

The proposed system not only determines the activity performed by the user, but the frequency at which it is performed. With this improvement in the activity recognition system, the information obtained can therefore be enriched with information specifying the number of pedal rotations, stairs, or steps that users perform per minute.

To achieve this degree of specification, a study of the accelerometry frequency component is made, whereby the maximum frequency module is used to determine the cadence of a range of activities. Therefore, the maximum frequency of the module could be useful in the analysis of the cadence of different activities, i.e. the max module, and at a second level, of the frequency associated with that value. These two values can be identified in Fig. 7 by the two grey dotted lines.

From a simple expression in conjunction with these values, the frequential value associated with the recognized activity in the current time window can be determined. It is denoted by af :

Table 2
Performance comparison by using measures of evaluation.

Activity	Measure									
	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
Climb	98.93	98.17	95.40	90.79	99.43	99.22	96.00	94.35	95.70	92.54
Descend	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

$af = \text{Max frequency module} \cdot 60s$

thus obtaining the cadence of activity per minute. This value is then displayed to the user in the application developed to determine both the intensity of the activity carried out and its repetitions.

3.3.3. Pseudocode

A pseudocode that summarizes the entire procedure more clearly is presented.

```

/* Inputs */
x = x1, x2, ..., xm;
S = S1, S2, ..., Sm
/* Discretization and Class Integration process */
W = {}, V = {}
for each Si in S do
    Wi = Ameva_Discretization (Si)
    Vi = Class_Integration (Si)
    W = W + Wi
    V = V + Vi
end for
/* Activity-Interval process */
U = {}
for each Vi in V do
    Ui = Activity_Interval (Vi)
    U = U + Ui
end for
/* Classification process */
values = {}
for j = 1 to l do
    aux = 0
    for p = 1 to m do
        i = interval (x, p)
        aux = aux + Ui[p][i][j]
    end for
    values = values + aux
end for
/* Output */
return getClass (getIndex (values, max (values)))

```

4. Method analysis

Once the basis of the developed activity recognition algorithm was set out, an analysis of the new proposal was performed. To this end, the new development was compared with a widely used recognition system based on neural networks. In this case, both learning and recognition was performed by continuous methods.

The test process was conducted on a Google Nexus One for a group of 10 users. Notably, the activity habits of these users were radically different, since 5 users were under 30 years old while the rest were older than this age. For this test, a document was delivered to each user so that a description of the activity

performed could be reported, together with its start time and end time.

In order to more accurately determine the real activities that the user is carrying out, a specific application on the device has been developed in which the user must enter the activity tracked. Those activities conducted during the test process that had not been previously trained, were dismissed in order to analyze the system accuracy.

Finally, the learning process of each activity recognized by the system consisted of its performance for a time of 6 min. As for the recognition process, users were followed over a period of 72 h.

From these values the real dataset is obtained,¹ which enables the accuracy and delay time, among other indicators, to be determined. However, it has been decided to apply this method to a public human-activity recognition dataset published by the Center for Machine Learning and Intelligent Systems (Anguita, Ghio, Oneto, Parra, & Reyes-Ortiz, 2012). This decision allows this work to be compared with other proposals through the same information repository.

First, after conducting the performance tests, it was found that, in the case of Ameva, the flow of information between the device and the server was much lower. This server is responsible for safeguarding the training data and recognized activities. As can be seen in Fig. 8, the traffic of data necessary was 4.7 KBytes for the system based on neural networks, and 0.6 in the case of Ameva. That is, the traffic of information was reduced by more than 70%. This constitutes an advantage in terms of a reduction in additional costs arising from excessive use of the data network.

Moreover, it is crucial to consider energy consumption and the processing cost of the system when working with the mobile device. In this case, after comparing the above methods, the conclusion is reached that the method based on Ameva reduces the computational cost of the system by about 50%, as can be seen in Fig. 9. The time needed to process a time window by using the Ameva-based method is 0.6 s. However, for methods based on neural networks, this was 1.2 s.

In addition to this benefit with the device, a survey from with a number of questions concerning the system impression was given to users at the end of the testing process. In general, the Ameva-based system scored much higher, due to the fluidity experienced by users while they were working with the device and while the activity recognition service was being executed. Furthermore, the response by the users indicated that the smartphone's temperature was considerably lower than when the neural network solution was employed.

Last but not least, some measures of the methods presented below are analyzed. Accuracy constitutes the most widely used metric for the measurement of the performance of learning systems. Nevertheless, it has been widely demonstrated that, when

¹ Available at <http://madeirasic.us.es/idinfor/wp-content/uploads/2014/02/activities.csv>.

the prior class probabilities differ greatly, these measures become inappropriate because they fail to consider misclassification costs, are strongly biased towards the majority class, and are sensitive to class skews (Daskalaki, Kopanas, & Avouris, 2006; Huang & Ling, 2005).

Finally, the definition of the measures used are:

- *Accuracy* is taken as the degree of veracity, while in certain contexts precision may mean the degree of reproducibility. It is defined here as $accuracy = \frac{TP+TN}{TP+TN+FP+FN}$.
- *Recall* is the fraction of relevant instances that are retrieved. It is defined as: $recall = \frac{TP}{TP+FN}$.
- *Specificity* measures the proportion of negatives which are correctly identified as such. It is defined as: $specificity = \frac{TN}{TN+FP}$.
- *Precision* is the fraction of retrieved instances that are relevant. It is defined as: $precision = \frac{TP}{TP+FP}$.
- *F-measure* (F_1) is a measure of the accuracy of a test. It can be interpreted as a weighted average of the precision and the recall, and is defined as: $F_1 = 2 \frac{precision \cdot recall}{precision + recall}$.

Note that TP and TN denote the number of positive and negative cases correctly classified, while FP and FN refer to the number of misclassified positive and negative examples, respectively.

Based on these definitions, Tables 1 and 2 present the test. It was conducted on 8 different activities in order to unify the results for all users. In both tables, differences between the two methods, RNA and Ameva, can be observed. Most values presented for each measure and activity in Table 2 show that the Ameva method performs markedly better than does the RNA, especially as regards precision. That is to say, the number of false positives in the Ameva method is lower than that using the RNA method, as can be observed in Table 1.

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 the temporal nature of the Immobile activity, results from these two activities present a high level of disturbance in contrast to other activities.

5. Conclusions and future work

As mentioned in Section 1, the number of research studies into activity recognition has increased in numerous domains in recent years. The insightful practical implications include, for example in elderly care the estimation of the quality of self-care and the monitoring of activities of daily living. Furthermore, activity recognition enables obesity to be better fought and improvements to be made toward a more active life in proactive healthcare and can also obtain the correlation of activities with moods, mood swings, and manic depression in psychiatry.

In the workplace, it enables maintenance staff and crowds to be tracked; and accountability in security/workflow monitoring to be managed; activity information to be shared in groups; and diaries and auto-filling journals to be managed for later accounting in memory support.

Efficiency and accuracy are two elements that must be taken into account when any activity recognition system is implemented on a mobile device. In this work, a recognition system based on discrete variables is presented whereby the Ameva discretization algorithm and a new classification system, based on this algorithm, are used.

By using this process to increase the recognition frequency, it has been possible to obtain a physical activity reading every 5 s and to publish these readings in the user activity log (Alvarez-Garcia, Ortega, Gonzalez-Abril, & Velasco, 2010).

This classification algorithm is in a simple unified form for multi-class cases, and in general has equal performance or outperforms RNA in terms of accuracy, recall, specificity, precision, and F-measure (F_1). It is also very fast because it is based on the Ameva discretization algorithm and a majority voting system which both have a very low processing time.

Furthermore, although it has yet to be tested with other datasets, the core of this algorithm remains free of dependence on the features of any recognition activity dataset, and is therefore applicable to any dataset.

The strengths of the system are: the high success rate for which it has been possible to achieve an average accuracy of 98% for the recognition of 8 different types of activities; and, the reduced computational cost associated to the processing of data during the recognition process, thanks to the inclusion of discrete variables.

However, the main problem of this system based on statistical learning lies in the number of activities that can be recognized. Working solely with accelerometer sensors reduces the number of detected activities and there is occasionally a strong correlation between the variables, as can be observed with the Immobile and Drive activities in Table 2. Moreover, the position of the mobile device is a weakness of the developed system since it must be placed at the hip which it is not a natural position (sometimes a belt is necessary). Other positions have been tested, but have not achieved good results.

This system is currently focused on the detection of falls in elderly people, so that it can transmit an alarm signal to the family and/or medical centre. It complements existing telecare services, such as those as offered by the Andalusian Regional Ministry of Equality, Health and Social Policy.

To this end, a new system that can recognize basic activities, such as immobile, walk, run, climb and descend, no matter where the mobile device is located is currently under development. This is expected to provide a major improvement in that the user will not have to worry that the mobile device is placed in the right position for the activities to be detected correctly, and therefore the user can move in a more natural way.

Acknowledgement

This research is partially supported by the project Simon (TIC-8052) of the Andalusian Regional Ministry of Economy.

References

- Altun, K., Barshan, B., & Tunçel, O. (2010). Comparative study on classifying human activities with miniature inertial and magnetic sensors. *Pattern Recognition*, 43(10), 3605–3620.
- Alvarez-Garcia, J., Ortega, J., Gonzalez-Abril, L., & Velasco, F. (2010). Trip destination prediction based on past GPS log using a hidden markov model. *Expert Systems with Applications*, 37(12), 8166–8171 <<http://dx.doi.org/10.1016/j.eswa.2010.05.070>>.
- Anguita, D., Ghio, A., Oneto, L., Parra, X., & Reyes-Ortiz, J. (2012). Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine. In *Ambient assisted living and home care* (pp. 216–223).
- Bhattacharya, A., McCutcheon, E., Shvartz, E., & Greenleaf, J. (1980). Body acceleration distribution and O₂ uptake in humans during running and jumping. *Journal of Applied Physiology*, 49(5), 881–887.
- Bicocchi, N., Mamei, M., & Zambonelli, F. (2010). Detecting activities from body-worn accelerometers via instance-based algorithms. *Pervasive and Mobile Computing*, 6(4), 482–495.
- Brezmes, T., Gorricho, J., Cotrina, J. (2009). Activity recognition from accelerometer data on a mobile phone. *Distributed computing, artificial intelligence, bioinformatics, soft computing, and ambient assisted living* (pp. 796–799).
- Choudhury, T., Consolvo, S., Harrison, B., Hightower, J., LaMarca, A., LeGrand, L., et al. (2008). The mobile sensing platform: An embedded activity recognition system. *Pervasive Computing*, 7(2), 32–41.
- Daskalaki, S., Kopanas, I., & Avouris, N. (2006). Evaluation of classifiers for an uneven class distribution problem. *Applied Artificial Intelligence*, 20(5), 381–417.
- Dernbach, S., Das, B., Krishnan, N. C., Thomas, B. L., & Cook, D. J. (2012). Simple and complex activity recognition through smart phones. In *2012 8th International conference on intelligent environments (IE)* (pp. 214–221). IEEE.

- Duong, T., Bui, H., Phung, D., & Venkatesh, S. (2005). Activity recognition and abnormality detection with the switching hidden semi-markov model. In *IEEE computer society conference on computer vision and pattern recognition* (Vol. 1, pp. 838–845).
- Forrester. (2011). Customer technology survey, Tech. rep., North American Technologies.
- Fuentes, D., Gonzalez-Abril, L., Angulo, C., & Ortega, J. (2012). Online motion recognition using an accelerometer in a mobile device. *Expert Systems with Applications*, 39(3), 2461–2465.
- Gonzalez-Abril, L., Cuberos, F., Velasco, F., & Ortega, J. (2009). Ameva: An autonomous discretization algorithm. *Expert Systems with Applications*, 36(3), 5327–5332.
- Gonzalez-Abril, L., Velasco, F., Ortega, J., & Cuberos, F. (2009). A new approach to qualitative learning in time series. *Expert Systems with Applications*, 36(6), 9924–9927.
- He, Z., & Jin, L. (2009). Activity recognition from acceleration data based on discrete cosine transform and SVM. In *IEEE international conference on systems, man and cybernetics* (pp. 5041–5044).
- Hong, Y., Kim, I., Ahn, S., & Kim, H. (2008). Activity recognition using wearable sensors for elder care. In *Second international conference on future generation communication and networking* (Vol. 2, pp. 302–305).
- Huang, J., & Ling, C. (2005). Using AUC and accuracy in evaluating learning algorithms. *IEEE Transactions on Knowledge and Data Engineering*, 17(3), 299–310.
- Kwapisz, J., Weiss, G., & Moore, S. (2011). Activity recognition using cell phone accelerometers. *ACM SIGKDD Explorations Newsletter*, 12(2), 74–82.
- Lepri, B., Mana, N., Cappelletti, A., Pianesi, F., & Zancanaro, M. (2010). What is happening now? Detection of activities of daily living from simple visual features. *Personal and Ubiquitous Computing*, 14(8), 749–766.
- Li, M., Rozgic, V., Thatte, G., Lee, S., Emken, B., Annavaram, M., et al. (2010). Multimodal physical activity recognition by fusing temporal and cepstral information. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(4), 369–380.
- Mathie, M., Coster, A., Lovell, N., Celler, B., Lord, S., & Tiedemann, A. (2004). A pilot study of long-term monitoring of human movements in the home using accelerometry. *Journal of Telemedicine and Telecare*, 10(3), 144–151.
- Morillo, L. S., Ramirez, J. O., Garcia, J. A., & Gonzalez-Abril, L. (2012). Outdoor exit detection using combined techniques to increase GPS efficiency. *Expert Systems with Applications*, 39(15), 12260–12267<<http://dx.doi.org/10.1016/j.eswa.2012.04.047>> .
- Paoli, R., Fernández-Luque, F., & Zapata, J. (2011). A system for ubiquitous fall monitoring at home via a wireless sensor network and a wearable mote. *Expert Systems with Applications*, 39(5), 5566–5575.
- Pawar, T., Chaudhuri, S., & Duttagupta, S. P. (2007). Body movement activity recognition for ambulatory cardiac monitoring. *IEEE Transactions on Biomedical Engineering*, 54(5), 874–882.
- Ravi, N., Dandekar, N., Mysore, P., & Littman, M. (2005). Activity recognition from accelerometer data. In *Proceedings of the national conference on artificial intelligence* (Vol. 20, p. 1541).
- Reddy, S., Mun, M., Burke, J., Estrin, D., Hansen, M., & Srivastava, M. (2010). Using mobile phones to determine transportation modes. *ACM Transactions on Sensor Networks*, 6(2), 13.
- Soria Morillo, L., Ortega Ramirez, J., Gonzalez-Abril, L. (2012). Aplicaciones contextuales en dispositivos móviles: Arquitectura para la mejora de la eficiencia energética, EAE. Spanish Academic Editorial.
- Wang, Y., Lin, J., Annavaram, M., Jacobson, Q., Hong, J., Krishnamachari, B., Sadeh, N. (2009). A framework of energy-efficient mobile sensing for automatic user state recognition. In *Proceedings of the 7th international conference on Mobile systems, applications, and services* (pp. 179–192).
- Ward, J. A., Lukowicz, P., Troster, G., & Starner, T. E. (2006). Activity recognition of assembly tasks using body-worn microphones and accelerometers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10), 1553–1567.
- Zappi, P., Lombriser, C., Stiefmeier, T., Farella, E., Roggen, D., Benini, L., Tröster, G. (2008). Activity recognition from on-body sensors: Accuracy-power trade-off by dynamic sensor selection. In *Wireless sensor networks* (Vol. 4913, pp. 17–33).