Pers Ubiquit Comput
DOI 10.1007/s00779-011-0415-z

ORIGINAL ARTICLE

Personalization and user verification in wearable systems using biometric walking patterns

Pierluigi Casale · Oriol Pujol · Petia Radeva

Received: 14 November 2010 / Accepted: 20 April 2011

Abstract In this article, a novel technique for user's authentication and verification using gait as a biometric unobtrusive pattern is proposed. The method is based on a two stages pipeline. First, a general activity recognition classifier is personalized for an specific user using a small sample of her/his walking pattern. As a result, the system is much more selective with respect to the new walking pattern. A second stage verifies whether the user is an authorized one or not. This stage is defined as a one-class classification problem. In order to solve this problem, a four-layer architecture is built around the geometric concept of convex hull. This architecture allows to improve robustness to outliers, modeling non-convex shapes, and to take into account temporal coherence information. Two different scenarios are proposed as validation with two different wearable systems. First, a custom high-performance wearable system is built and used in a free environment. A second dataset is acquired from an Android-based commercial device in a 'wild' scenario with rough terrains, adversarial conditions, crowded places and obstacles. Results on both systems and datasets are very promising, reducing the verification error rates by an order of magnitude with respect to the state-of-the-art technologies.

Keywords User personalization · User verification · Wearable devices · One-class classification

1 Introduction

Mobile devices are nowadays multi-functional wearable computer systems. Their computational capabilities allow their use as GPS localization systems, portable video game systems or personal digital assistants. Their rapid acceptance in our modernized society is making these systems to grow oriented to on line shopping and consumer applications. While now these devices manage information about our position and the movements of the user, in the near future, they will manage private and sensible information such as our bank account and credit cards numbers. Protecting the device from illicit and inappropriate use needs to be ensured while making the authentication task as transparent to the user as possible. In this sense, unobtrusive biometric measures become appropriate tools for verifying user identity. Biometrics are currently available on mobile devices. Modern smart phones and PDA have integrated accelerometers able to detect changes in the orientation and acceleration of the device and, consequently, of the person wearing it.

Accelerometers have been widely accepted by the scientific community due to their miniaturization, their low-power requirement and for their capacity to provide data directly related to motion. Many research works [1, 10, 13] use accelerometers placed on the body for recognizing everyday life activities or personal posture. In [10], authors give a complete review about the state of the art of activity classification using data from one or more accelerometers. Recognizing physical activities has been also addressed under practical and commercial perspective. In [9], authors

P. Casale (✉)
Computer Vision Center, Barcelona, Spain
e-mail: pierluigi@cvc.uab.es

O. Pujol · P. Radeva
Departamento de Matematica Aplicada i Analisi, Computer Vision Center, Universitat de Barcelona, Barcelona, Spain
e-mail: oriol_pujol@ub.edu

P. Radeva
e-mail: petia.ivanova@ub.edu

Published online: 08 July 2011

summarize their experience in developing an automatic physical activities recognition system and their results are of very practical interest. Physical activity recognition represents the starting point to make biometric measurement suitable to verify authorized users.

To the best of our knowledge, user verification by means of accelerometer data has been rarely addressed. User verification is usually measured in terms of the false acceptance rate (FAR) and false rejection rate (FRR). FAR measures the probability of an unauthorized user to be confused with a legit user. On the other hand, FRR measures the probability of the system to misclassify an authorized user as a non-authorized one. While the first measurement concerns the robustness of the system to intruders, the second measurement regards usability and inconspicuousness of the system. In [17], a walking-based authentication system has been integrated with fingerprints data and voice recognition, ensuring in this way an high degree of reliability. Walking-based authentication provides 0.14 of equal error rate between FRR and FAR with little variations if the system is brought on the chest or on the waist. In this work, data are acquired from an ad hoc accelerometer-based system assembled for the experiments. Even if in this work authors do not use a real mobile phone, this represents the first work using walking biometrics for user verification. Some efforts have been done also in [5], where, using the accelerometer of an Android-based mobile phone, walking data have been collected from 51 testers walking for two runs in an interior corridor. Authors report results of 0.2 of equal error rate. Results obtained are encouraging but not persuasive to be implemented in a mobile device for a reliable system. That work is the first one where user verification has been done using an everyday current mobile phone. A close notion to user verification concerns user identification – not necessarily needed for authentication. In [8], a biometric user-authentication system based on a person gait is proposed. Applying histogram similarity and statistics about walking-cycle, authors ensure 0.16 of identification error rate.

There seems to be a physiological justification to the fact that persons can be distinguished on the base on their walking style. In [3], authors state that there is inter-individual variability in walking styles between persons particularly at moderate and fast velocity. They show that this variability cannot be simply explained on the basis of the different biomechanical characteristics of the subjects, but that it depends on the different kinematic strategies. Subjects differ in their ability to minimize energy oscillations of their body segments and to transfer mechanical energy between the trunk and the limbs. Individual characteristics of the mechanical energy expenditure were correlated to the corresponding kinematic characteristics. Accordingly to this, in [16], using a GPS sensor, authors are able to capture basic gait parameters over a period of time of 5 s. They observed a specific gait pattern for slow walking and the walking patterns in free-living conditions exhibit low intra-individual variability, but that there is substantial variability between subjects.

In this work, a discriminative machine learning pipeline for user verification with personalization is proposed. Discriminant classifiers have proven to be extremely efficient and powerful tools, even surpassing the performances of generative machine learning techniques. Using this framework, a two-stage process is defined. Starting from a general physical activity classifier based on AdaBoost [7], trained on a baseline training set, the system is personalized adding data of walking activities of authorized users in order to boost the classification performances for those users. Since AdaBoost is an incremental classifier, this process is extremely efficient because just further weak classifiers need to be added to the original baseline classifier.

Based on this personalized system able to filter in its first stage many walking activities of unauthorized users, authorized users are also verified. Modeling the walking activity of authorized users is considered a one-class classification problem [15]. In one-class classification, the boundary of a given dataset is found without any counter examples. This is the case of user verification since one cannot explicitly provide examples for modeling all possible non-authorized users. Thus, in a second stage, an ensemble of one-class classifiers is created using the concept of convex hull. A four-layer architecture is introduced in order to provide the convex hull with improved features. The inner most layer concerns a computationally efficient way of building the convex hull in multiple dimensions. Since the convex hull structure is very sensible to outliers, the next layer concerns a robust way to ignore outliers and defines the core distribution of the data. The third layer extends the convex hull to model non-convex shapes using a mixture model. Finally, the fourth and last layer considers the temporal coherence of the accelerometer data stream to improve the results.

This novel technique is validated in two different environments with two different wearable systems. First, a custom wearable system is built and used in a general activity scenario. Second, a commercial mobile device is used to verify users ‘in the wild’ such as in rough terrains or in adversarial conditions, in crowded places or with obstacles. Results prove that the verification system performs well on both verification accuracy and computational cost points of view. Results obtained show that users can be verified with high confidence, with very low false acceptance rate and false rejection rate, decreasing the best results reported by an order of magnitude.

The proposed techniques, even being truly general, may be used in many other applications besides the mere authentication task. Many applications can be found in the fields of ambient intelligence and pervasive computing where a system, like the one proposed in this paper, can provide a continuous authentication mechanism and be complemented by other further authentication mechanisms able to provide very high and reliable performances in critical situations. Consider, for instance, intelligent and personalized settings of domestic and working environments. If a wearable sensor might be able to constantly authenticate you, the smart home environment always would provide you personalized services and attention according to your needs. Analogously, at work, the intelligent environment might provide a tracking of the to do list of the day. Many others applications might be found in health-care field, like geriatric environments or, even, in the post-operative recovery of patients using clinical monitoring devices such as holter-like equipment.

The article is structured as follows. Section 2 introduces the general user verification system and describes the details of the personalized action recognition and user verification subsystems. Section 3 describes the experiments and results for each subsystem using two wearable devices: a custom wearable device and a commercial mobile phone system. In Sect. 4, we discuss different interesting questions regarding the verification process, such as the influence of the non-convex mixture model or the effect of the temporal ensemble. Finally, Sect. 5 concludes the article.

2 User verification system

A two-stage pipeline is proposed for user verification with personalization. The overall user verification system is shown in Fig. 1. The first stage consists of a personalized activity classifier. As a result of the first stage, only data belonging to the class ‘walking’ is provided as input to the user verification stage. This second stage is tuned to verify the walking biometric parameters of an authorized user.

2.1 Personalized activity recognition subsystem

The underlying idea behind the personalization step in this subsystem is to bias a general activity recognition classifier toward the data of authorized users. Thus, a general activity recognition is trained to distinguish among different general daily activities such as, walking, climbing stairs, standing, working seated and interacting with the environment (which includes vending machines, ATM or talking with other users). This classifier is trained using data from a general set of people performing these activities.

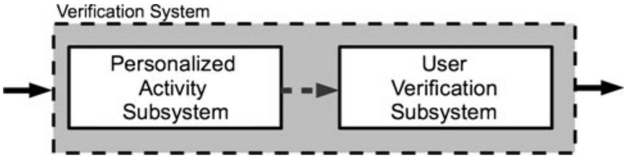


Fig. 1 Block diagram of the user verification system

The general activity classifier is based on a multi-class extension of AdaBoost [7]. It receives as input the features extracted from accelerometer data and it detects when walking activities occur. AdaBoost is an efficient incremental algorithm for supervised learning. AdaBoost boosts the classification performance of a weak learner by combining a collection of weak classification functions to form a strong classifier with high performance. The algorithm combines iteratively weak classifiers by taking into account a weight distribution on the training samples such that more weight is given to samples misclassified in previous iterations. The final strong classifier is a weighted combination of weak classifiers followed by a threshold.

Table 1 shows the pseudo-code for AdaBoost. The algorithm takes as input a training set $(x_1, y_1), \ldots, (x_m, y_m)$ where x_k is a N-dimensional feature vector, y_k are the class labels and $D_1(k)$ an uniform weights distribution over the training examples. At the training step t, a weak classification hypothesis h_t is selected with error $\epsilon_t \geq 0.5$. The weight α_t correspondent to the current hypothesis h_t is computed proportional to the error ϵ_t. Examples are weighted based on the updated distribution proportional to the current hypothesis. In this way, misclassified examples will have, in the next step, more weight than well-classified examples. After T rounds of training, the weak classifiers h_t and ensemble weights α_t are used to assemble the final strong classifier.

The multi-class extension of AdaBoost is performed using an Error Correcting Output Codes One-Vs-All [6] technique.

For classifying activities, AdaBoost is previously trained using a large amount of data from many subjects as a general activity recognition classifier $H_{\text{original}}(x)$. However, when one applies this general classifier to a specific user, since this user data have never been seen by the system, its performance may be poor. For this reason, when the system is given to an authorized user, a dataset X_{auth} from this user is recorded. Due to the inherently incremental nature of AdaBoost, it is possible to perform T runs of training using a dataset and to follow the training process for T' runs using a new dataset. This feature becomes crucial in the personalization step of the proposed pipeline. In order to bias the performance of the general classifier toward an authorized user, the recorded data from this user X_{auth} are used for T' runs. This adds to the previous strong classifier T' new weak classifiers that takes into account the specific biometric features of the authorized user and specializes on

Table 1 AdaBoost Algorithm

```
- Given a training set $(x_1 , y_1), \ldots, (x_m, y_m)$, with $x_k \in R^N$, $y_k \in Y = \{ -1, +1\}$;
- Initialize weights $D_1(k) = 1/m$, $k = 1, \ldots, m$;
- For $t = 1, \ldots, T$:
    1. Train weak learner using distribution $D_t$
    2. Get weak hypothesis $h_t$: $X \rightarrow \{ -1, +1\}$
            with error $\epsilon_t = Pr_{k \sim D_t}[h_t(x_k) \neq y_k]$
    3. Choose $\alpha_t = \frac{1}{2}\ln(\frac{1-\epsilon_t}{\epsilon_t})$
    4. Update :
        $D_{t+1}(k) = \frac{D_t(k)\exp(-\alpha_t y_k h_t(x_k))}{Z_t}$
       where $Z_t$ is a normalization factor chosen
       so that $D_{t+1}$ will be a distribution.
- Output the final hypothesis
        $H(x) = sign(\sum_{t=1}^{T} \alpha_t \, h_t(x))$
```

them. Observe that, since our biometric verification system from inertial data is based on the walking activity, only data that the general classifier labels as ‘walking’ are used for this specialization. As a result of this personalization, we expect that the performance when classifying authorized users walking activities will be enhanced for the specific user and a big load of walking activity from other users considerably filtered. Thus, this first classifier serves two purposes. First, it filters walking activity from the rest of usual daily activities. And second, due to the personalization process, it filters many walking activities from non-authorized users.

2.2 Verification subsystem

Once inertial data are selected and filtered by the personalized activity recognition system, the system must verify if the walking data obtained belong to a registered user or not. This task is done by the verification subsystem. Verification is sometimes confused with recognition. However, while in recognition one has to chose among different possible choices or classes, in verification one must just decide if data belong to a given class. One may argue that the verification task can be reduced to that of recognizing the desired class versus the non-desired one. Although there are some scenarios in which this is true, this reduction does not hold in general. This is because a recognition system needs to correctly model all the different choices in order to achieve good results. However, the class *non-verified user* cannot be effectively modeled since it would require the knowledge of all possible users. For this reason, probabilistic approaches that model the distribution of just one class are usually used. If one follows this line of work, the discriminative counterpart of the aforementioned problem is called *one-class classification* and this is focused on finding the boundaries of the desired class. One of the most successful strategies and state-of-the-art in one-class classification is one-class Support Vector Machine (SVM) [15]. However, training an SVM is computationally expensive and it cannot be done efficiently in embedded systems or mobile phones not only due to the computational complexity but also because it usually involves a delicate parameter tuning step.

In this work, we follow the guidelines of one-class classification strategies and propose a fast, efficient and effective approach. At first glance, the method may seem rather involved. Thus, in the following lines, we explain the different parts of the system.

The verification system is structured in layers (see Fig. 2). The inner most layer concerns the way of modeling the one-class problem. The approach proposed is based on the convex hull geometric structure. The convex hull of a set of points is defined as the smaller polytope such that all elements are enclosed in it. Additionally, for all pair of elements inside the convex hull, any line segment that joins them must be inside or belong to the convex hull. Observe that the convex hull models the boundary of a set of points. If this set of points belongs to the authorized user, the process of verifying whether new data belong to an

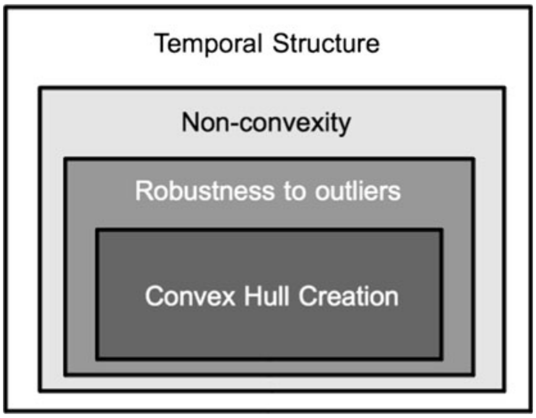


Fig. 2 Layer structure of the verification subsystem

authorized user or not is reduced to the problem of knowing if these data lie inside or outside the convex hull. The use of the convex hull is theoretically justified in [2] where authors state that, in binary classification, finding the maximum margin between two classes is equivalent to finding the closest points in the convex hull delimiting the classes. Note that this is equivalent to use an SVM classifier if classes are linearly separable. In this work, we propose the notion of convex hull for modeling one-class problems which, to the best of our knowledge, has never been done before.

The different layers created around this problem deal with the concept of robustness and appropriateness of the convex hull for modeling user data. The inner most layer builds a convex hull efficiently. The second layer concerns the robustness of the convex hull to outliers by means of a bagging strategy. The third layer models the problem of the convex hull to approximate non-convex shapes. Although user walking data seems to be well defined by a convex hull, if one looks at different runs of the same user, one may observe that there is so much variability that makes the problem much better fitted by non-convex polytopes. Thus, one convex hull does not suffice. Finally, the last layer considers the temporal coherence in the walking signal of the user in order to reduce the verification error rates. In the following subsections, the different algorithms for tackling these problems are described in detail.

2.2.1 Layer 1: initial convex hull

A one-class classifier ensemble has been trained using the convex hull generated on a low-dimensional features space. The underlying idea is shown in Fig. 3 where a scatter plot of two features are reported. In this low-dimensional features space, data related to each user are localized in a specific region of the features space. Training the one-class classifier means building the convex hull and defining the region of the space where user data lie. When a new point appears, if the point is inside the convex hull, then it represents a walking activity of the user.

The creation of the convex hull is computationally intensive in general and requires $\mathcal{O}(N^{\lfloor d/2 \rfloor + 1})$, where d is the dimensionality of the data and N the number of data examples. Even worse, the memory usage needed for storing all data points that create the facets of the convex hull may be arbitrarily high. Since this cost is prohibitive in time and memory and we only need to check if a point lies inside the convex hull, we may use a set of k two-dimensional projections of the data, build the convex hull in the two-dimensional case and check if there exist any projection in which the testing element is outside the projected convex hull. If it does, then the point effectively lies outside the original d-dimensional convex hull. Figure 4 shows an example of a 3D convex hull, a test point outside of the hull and three candidate projection planes. At the bottom of the figure, we may observe that in two of those projections the projected test data are outside the projected convex hull. Table 2 describes the pseudo-algorithm for creating the convex hull approximation and testing if a point lies inside the hull.

This projection approach has different important computational and storage advantages. On one hand, given a training set of N examples, the computational cost of building the convex hull in a bi-dimensional space is $\mathcal{O}(N \log N)$. Let K be the number of examples that define the convex hull – the set of examples of the dataset that conform the facets of the polygon – then $K \ll N$ and the cost of testing if a data point lies inside or outside the convex hull is $\mathcal{O}(K)$. Another worthwhile advantage of computing the convex hull is that it can be built or updated online – as new training data arrives – with cost $\mathcal{O}(\log N)$ [12]. Thus, if we use t projections, the final computational cost for building this approach is $\mathcal{O}(tN \log N)$ and the test cost $\mathcal{O}(tK)$.

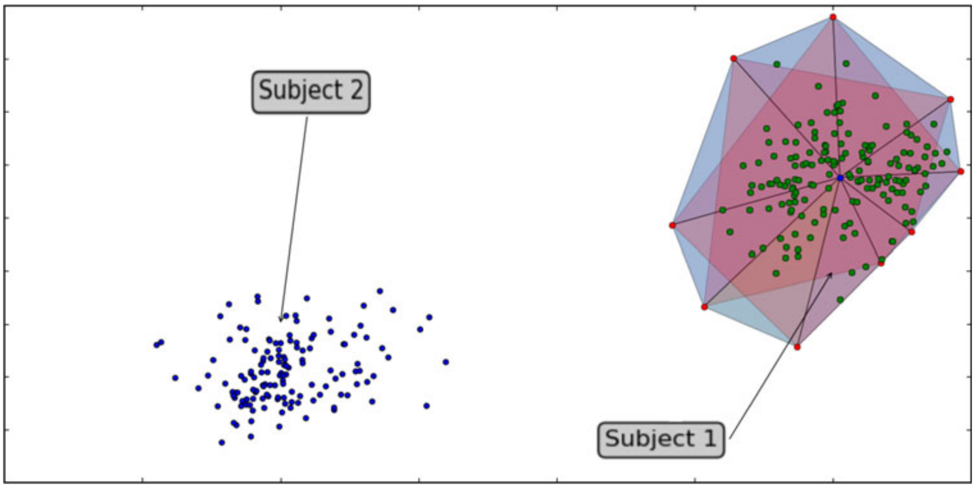


Fig. 3 User verification using a convex hull as one-class classifier

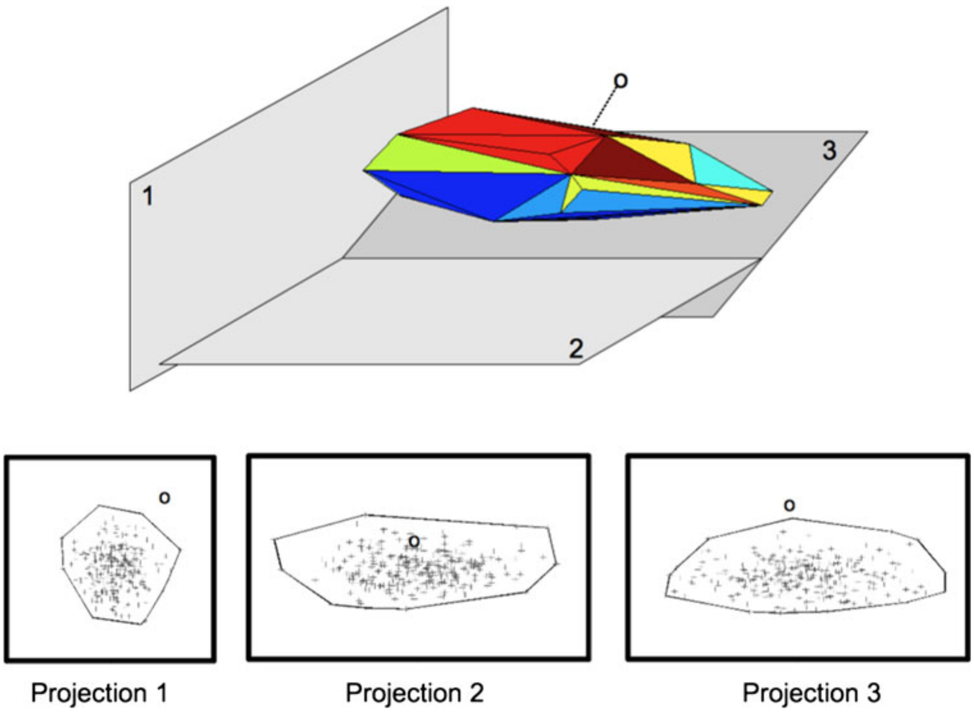


Fig. 4 Layer 1: convex hull and several projections. A point is outside of the convex hull if and only if there exist a projection in which the point is outside of the projected convex hull

Table 2 Approximate convex hull test building

```
–Train:
- Given a training set $X \in \mathcal{R}^M$, where M is the number of features
    1. For each pair of vectors $(v_i, v_j)$ spanning the $\mathcal{R}^M$ space
      2. Project data into the selected subspace.
      3. Compute the convex hull $CH_{\{i,j\}}$ for the
           projected dataset
  Return: the set $\{CH_{\{i,j\}}\} \quad \forall\, i, j$
–Test:
- Given a point $p \in \mathcal{R}^M$;
- Given a set of convex hulls $\{CH_{\{i,j\}}\} \quad \forall\, i, j$
    1. For each convex hull in $\{CH_{\{i,j\}}\}$
      2. Project $p$ into the subspace defined by vectors $v_i, v_j$.
      3. Find the barycenter $b$ of the polygon.
      4. For each point $c$ of the convex hull.
        5. If $c \notin CH_{\{i,j\}}$
          6. return OUTSIDE
    7. return INSIDE
```

2.2.2 Layer 2: Robust convex hull

A question that needs to be addressed when using a convex hull is its robustness with respect to outliers. Convex hulls are very sensitive to outliers and outliers can heavily influence the performance of the verification process. If we assume that data from accelerometers are noisy and can contain outliers, the resulting convex hull will not represent user data accurately. In order to reduce the influence of outliers, we propose to use a bagged set of convex hulls. Bagging [4] is a machine learning ensemble meta-algorithm that improves classification models in terms of stability and classification accuracy. It reduces variance and helps to avoid overfitting.

Given a training set $X \in \mathcal{R}^M$ of size N, bagging generates l new training sets X'_i of size N by sampling examples from X uniformly with replacement. Then, l models (convex hulls) are trained using the examples of each subsampled training set X'_i and combined by majority voting.

In the scatter plot shown in Fig. 3, described in the previous section, the result of building three convex hulls on different subsampled sets is shown. The result of this process reduce the influence of the elements on the boundary of each convex hull and better defines the core set of examples of the training set.

2.2.3 Layer 3: mixture of convex hulls

As we argue before, data from different walking runs of one user can be very different and not necessarily be well represented by a convex shape. In order to overcome this drawback, a mixture of convex hulls is built in this stage (see Table 3). The mixture begins with one convex hull. As data are recorded, the performance of this convex hull is checked. If data are well represented by the current model, there is no need to change anything. But if a stream of data is not correctly represented by the model, a new convex hull is created and added to the user profile model (Table 3).

Table 3 Mixture of convex hulls algorithm

```
–Train:
- Given a training source that produces streams of data $X_i \in \mathcal{R}^M$;
- Given a layer-2 convex hull creation function such that
  $CH = f_{CH}(x)$;
- Given a user model $\mathcal{M}$ formed by layer-2 convex hulls and a test
  function $t_{\mathcal{M}}(x)$;
  1. While not end of training
    2. If $t_{\mathcal{M}}(X_i) < \tau$ %% $\tau$: Performance measure
      3. $\mathcal{M} = \mathcal{M} \cup f_{CH}(X_i)$
  Return: $\mathcal{M}$
```

2.2.4 Layer 4: temporally coherent convex hulls

The final step is to take into account the temporal coherence of each stream of user data. The assumption in this layer is that in a given temporal window, the user of the device does not change, and thus, considering the full length of the window, he/she must be either an authorized or an unauthorized user. Thus, we propose a simple majority voting on a temporal window of walking data. That is, given a sequential training set $X = \{x_1, x_2, \ldots, x_N\}$, a sliding window of size t is used on the predictions of the layer-3 convex hull $\bar{y}$ of those examples. As a result, given a sequence of predictions, the final decision is achieved by $\hat{y}_i = majority\{\bar{y}_{i-t}, \bar{y}_{i-t+1}, \ldots, \bar{y}_i\}$.

3 Experiments and results

In this section, an exhaustive validation of the verification system is provided. The section is divided in three parts: first, we comment the features computed from the accelerometer data. These features are common in both scenarios described in the following subsections. The next two subsections are devoted to the validation of the personalized activity recognition subsystem and the validation of the user verification subsystem. In each subsection, experimental settings, validation protocols, data acquisition conditions and systems as well as the results for each subsystem are described.

3.1 Feature computation

Measures related to the variations of the oscillations in the acceleration waveform of the tri-axial accelerometer have been computed as follows:

- difference between pairs of consecutive peaks;
- difference between the value of consecutive upper-side peaks;
- difference between the value of consecutive lower-side peaks.

The first quantity provides important informations about the variation of the intensities in the acceleration during an activity. The second and third features provide informations about the shape of the waveform. A further time series has been obtained computing the derivative of acceleration data. The derivative of acceleration, called *Jerk* [14], represents the rate of change of the force acting on a body. From this new signal, the same features have been extracted.

Using those measures, the oscillatory movements typical of different activities and, at the same time, typical of the interpersonal differences are taken into account. On one side, the mean value of those quantities, computed for both acceleration and jerk, is used for activity classification, yielding an eighteen-dimensional feature vector. The three measures previously exposed are computed for both acceleration and jerk time series on each of the three acceleration axis provided by the tri-axial accelerometer yielding a total of 18 different measurements. On the other hand, for verification purposes, the standard deviation of these quantities is computed and stored into a different eighteen-dimensional feature vector.

3.2 Personalized activity recognition results

3.2.1 Experimental settings and validation protocols

3.2.1.1 Data acquisition system A custom wearable system has been designed for multi-sensors data acquisition and processing. The core of the system is the BeagleBoard, a low-power, low-cost single-board computer with a Linux embedded operating system compiled ad hoc for the board. The system acquires audio and video by mean of a cheap web-cam and motion data by means of a accelerometer transmitting data over a bluetooth connection. The system can be easily brought in one hand or worn in a little bag around the waist. The sensors can be placed all in the same part of the body or localized in different parts. For these specific experiments, sensors have been localized on the chest. Accelerometer data are sampled at 52 Hz with a resolution of ±4 g.

Figure 5 shows the system disassembled on a table, showing its components.

Data have been collected from ten volunteers, three women and seven men with age between 27 and 35. Given the small form factor of the device, users were able to perform activities in the environment they selected in complete autonomy and for a minimum time of 5 min. In this way, the

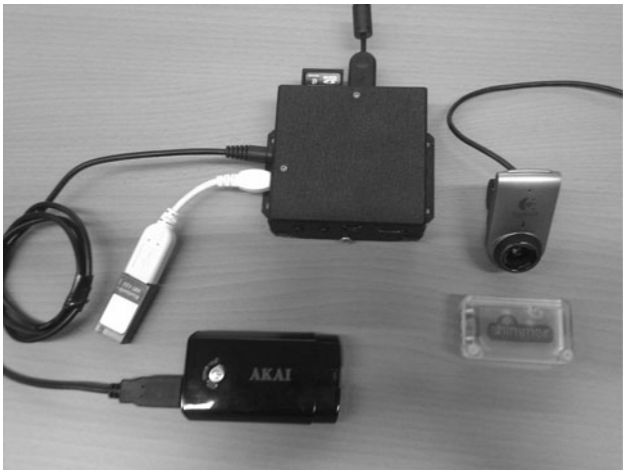


Fig. 5 The wearable system BeaStreamer-0.1

laboratory setting limitation is fully overpassed and a total amount of 7 h and 11 min has been collected. In the following, the list of activities performed with the time amount of data collected is shown:

- climbing stairs: 60 min
- walking: 103 min
- interacting with environment: 120 min
- standing: 54 min
- working at computer: 90 min

Ground truth is provided automatically by the system that labels activities with a sequential number providing the duration of the activity performed, too. When the users perform one of the defined activities, they just need to press the power button of the system and the acquisition process starts automatically. They can stop the acquisition process pressing again the start button. In this way, we free the users from the tedious task of labeling activities and taking care of annotating the duration of each activity. We just ask users to annotate the sequential order in which they perform the activities.

In the preprocessing stage, data have been smoothed with a moving average smoothing filter. For activity classification, data have been discretized using a sliding window technique using temporal windows of 2 s.

3.2.1.2 Performance measurements Accuracy, precision and recall have been chosen as classification performance measures. The quantities are defined in Eqs. 1–3 in terms of the elements of the confusion matrix C. An element $C(i, j)$ of the confusion matrix accounts for the number of examples of class j classified as class i. Thus, a confusion matrix where $C(i,i) \neq 0$ and $C(i, j) = 0$, $i \neq j$ shows a perfect classification results.

$$\text{Accuracy}_i = \frac{C(i,i) + \sum_{l \neq i, m \neq i} C(l,m)}{\sum_{i,j} C(i,j)} \tag{1}$$

$$\text{Precision}_i = \frac{C(i,i)}{\sum_{j \neq i} C(j,i)} \tag{2}$$

$$\text{Recall}_i = \frac{C(i,i)}{\sum_{j \neq i} C(i,j)} \tag{3}$$

3.2.1.3 Validation protocol In order to obtain performance measurements, the general activity classifier has been trained and validated using a Leave-One-User-Out (LOUO) cross-validation algorithm. In LOUO, each subject is used once for testing on a classifier trained on the rest of users. This process is repeated for each user, and the performance measurements are computed.

In order to validate the personalized version of the activity classifier, a modification in the LOUO protocol has been used. Since the algorithm needs a small set of data from the user to be verified for the classifier personalization step, we use a mixed validation approach of N-fold cross-validation and leave-one-user-out cross-validation. In this protocol, data belonging to each user is randomly split in N equal size, non-overlapping subsets, namely folds. This folded version of LOUO iterates over both subjects and folds. In this way, all the examples of all the subjects are used exactly once for testing.

3.2.2 Experimental results

Using the LOUO cross-validation and experimental settings described in the former section, the general activity recognition system performances are shown in Table 4.

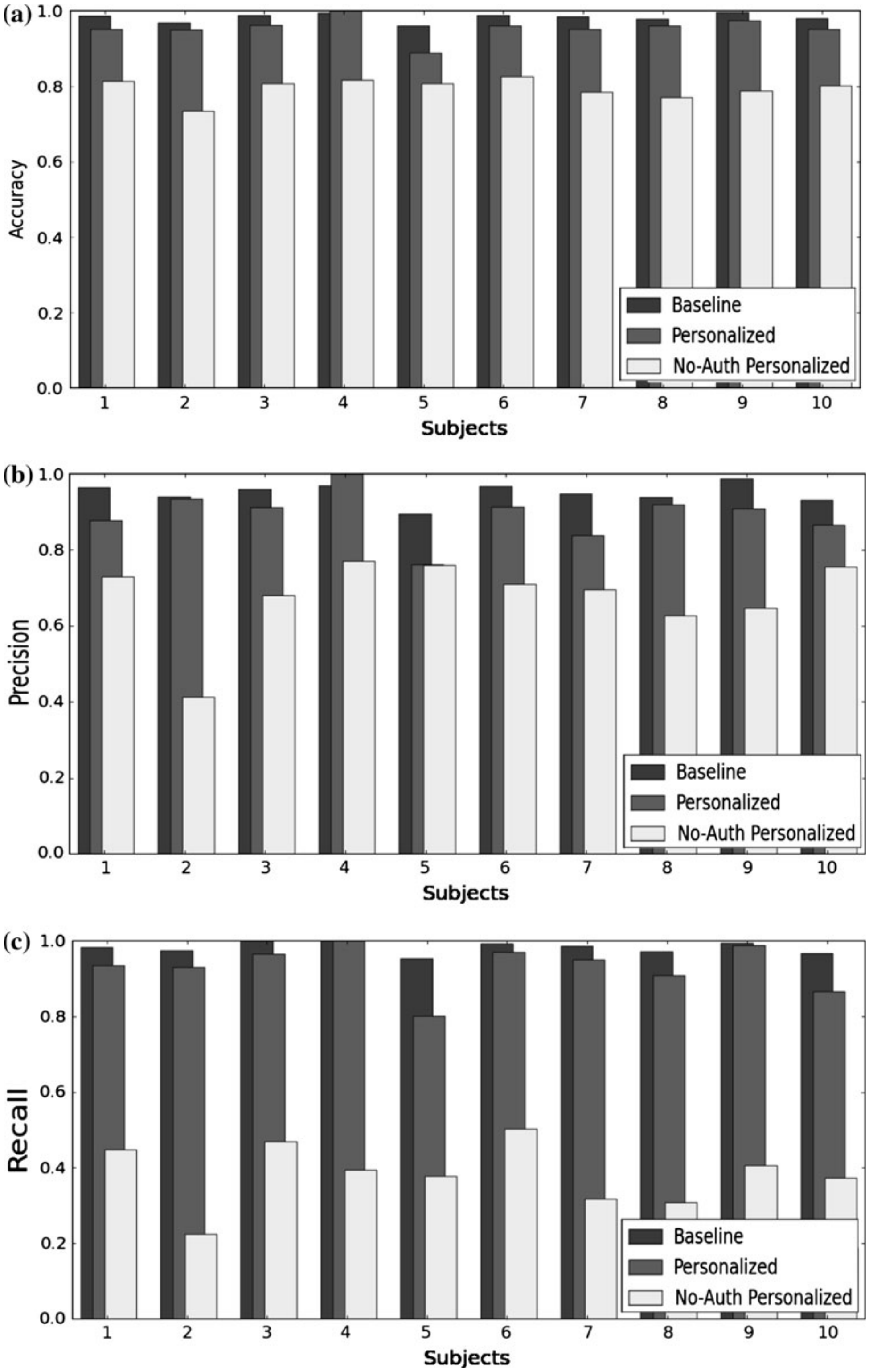
All the activities are classified with good performances. Observe that *walking* is the activity having the best overall performances for all the metrics taken into account. This effect is expected since the features extracted are specifically designed to capture subtleties in the walking pattern.

The effect of personalization is shown in the following figures. Accuracy, precision and recall for the baseline classifier and personalized classifier are shown in Fig. 6.

Table 4 Classification performances for general activity classifier

	Accuracy (%)	Precision (%)	Recall (%)
Climbing stairs	94.26 ± 2.1	99.6 ± 0.25	88.82 ± 4.1
Standing	72.53 ± 4.7	96.57 ± 1.3	46.68 ± 10
Talking	84.96 ± 6.4	94.01 ± 3.1	74.6 ± 12.5
Walking	97.65 ± 2.2	97.98 ± 1.3	97.28 ± 3.3
Working	87.1 ± 5.3	94.77 ± 2.1	78.6 ± 11.3

Fig. 6 **a** Accuracy of general walking classification, personalized classification and non-personalized classification. **b** Precision of general walking classification, personalized classification and non-personalized classification. **c** Recall of general walking classification, personalized classification and non-personalized classification



Black bars represent performances obtained testing the baseline classifier on the subject, gray bars represent performances obtained testing the personalized classifier on the user and white bars represent performances obtained testing the personalized classifier on all the other subjects. Personalization ensures that the performances of the classifier are drastically reduced for all the subjects except for the user.

3.3 User verification subsystem results

3.3.1 Experimental settings and validation protocols

3.3.1.1 Data acquisition system The validation of this subsystem has been performed using two different acquisition systems and conditions. First, the same acquisition system and data as in the validation of the personalized activity recognition subsystem are used. However, in order to have a more complete experimental section, we created an additional database using a different acquisition system with different experimental settings using an Android-based mobile phone.

Acceleration data from walking activities have been acquired using a Google Nexus One mobile phone with operating system Android 2.2. Android-based mobile phones have an open Application Program Interface that allows programming the phone and accessing the sensors present in it. In this way, it is possible to read and save accelerometer data. Every sensor in the Android platform has an listener associated that delivers data when a change in its value happen. The delivery rate can be set to different frequency but this value is just an hint to the system. Events may be received faster or slower than the specified rate. In the setting for the experiments, accelerometer has been sampled with a timestamp of approximatively 33 ms, using the mobile phone in normal mode, with all the network connections active.

Data have been collected from 20 testers with ages between 25 and 35, 15 men and 5 women. Testers perform seven different runs of walking, for a total of 140 different walking runs. Testers were free to perform all the runs as they like. The mobile phone has been put in the jacket pocket on the chest. The acceleration axis are concordant to the specification of the Android platform and, in our setting, the Z axis refers to the direction concordant to the movement. The walking activity is performed in indoor, outdoor and urban environment. The walking scenarios are described in the following:

- 1: Indoor corridor;
- 2: Outdoor street uphill and downhill;
- 3: Crowded flat urban street;
- 4: Free flat urban street;
- 5: Mixed scenario: Passing through doors from urban environment to indoor environment with people;
- 6: Mixed scenario: Semi-Indoor corridor with up and down ramps;
- 7: Walking in a garden with rough floor.

In all scenarios, data have been discretized using sliding temporal windows of 5 s, with 25% of overlapping between windows.

3.3.1.2 Performance measurements False Rejection Rate (FRR) and False Acceptance Rate (FAR) have been chosen as performance measures for verification. The FRR is defined as the percentage of identification instances in which false rejection occurs or the percentage of how many times an authorized user is not well verified. FAR is a measure of the likelihood that the system will wrongly accept an unauthorized user.

3.3.1.3 Validation protocol In order to validate the verification subsystem, we use a 10-Fold cross-validation strategy. Since we are in front of a one-class problem, data from only one user are used for training purposes. Thus, we use onefold of a user for training purposes and the rest of the data of that user and all data from all other subjects for testing. This process is repeated for each fold and user.

3.3.2 Experimental results

3.3.2.1 Custom wearable system results A layer-4 verification system is used in a temporal frame of 55 s—the verification result is achieved after 55 s observing user data. In the discussion section, more details on the performance when varying this temporal frame will be given.

Figure 7a, b picture the box plot showing the distribution of the cross-validation results for each user for both values FAR and FRR, respectively. Observing the FAR box plot, most users achieve a FAR value below 10^{-2} with the exception of two outliers. This same effect can be seen in the FRR plot, with most users with FRR values below $5 \cdot 10^{-3}$. The final ensemble achieves a FRR mean value of 0.0072 ± 0.0025 and a FAR mean value 0.03 ± 0.07. Observe that in this scenario, the rejection of an authorized user is smaller than the acceptance of an intruder. This means that the system may block an authorized user with very small probability. This feature is important for usability purposes, since we want the system to be as unobtrusive as possible.

3.3.2.2 Android-based system results The results in a wilder scenario are evaluated using data collected with the Android-based mobile phone with the same temporal frame. In Fig. 8a, b FAR and FRR distributions for each subject are shown in box plots. Observe that for the great majority of users FAR is below $5 \cdot 10^{-4}$. There is one pathological example in which its FAR is over 10^{-2}. The false rejection rate is a little worse but the majority of users FRR is below $2 \cdot 10^{-2}$. The average FAR obtained is 0.0011 ± 0.0023, and the average FRR is 0.020 ± 0.0352. In this scenario, the user is very well discriminated from the rest of users but its acceptance is lower.

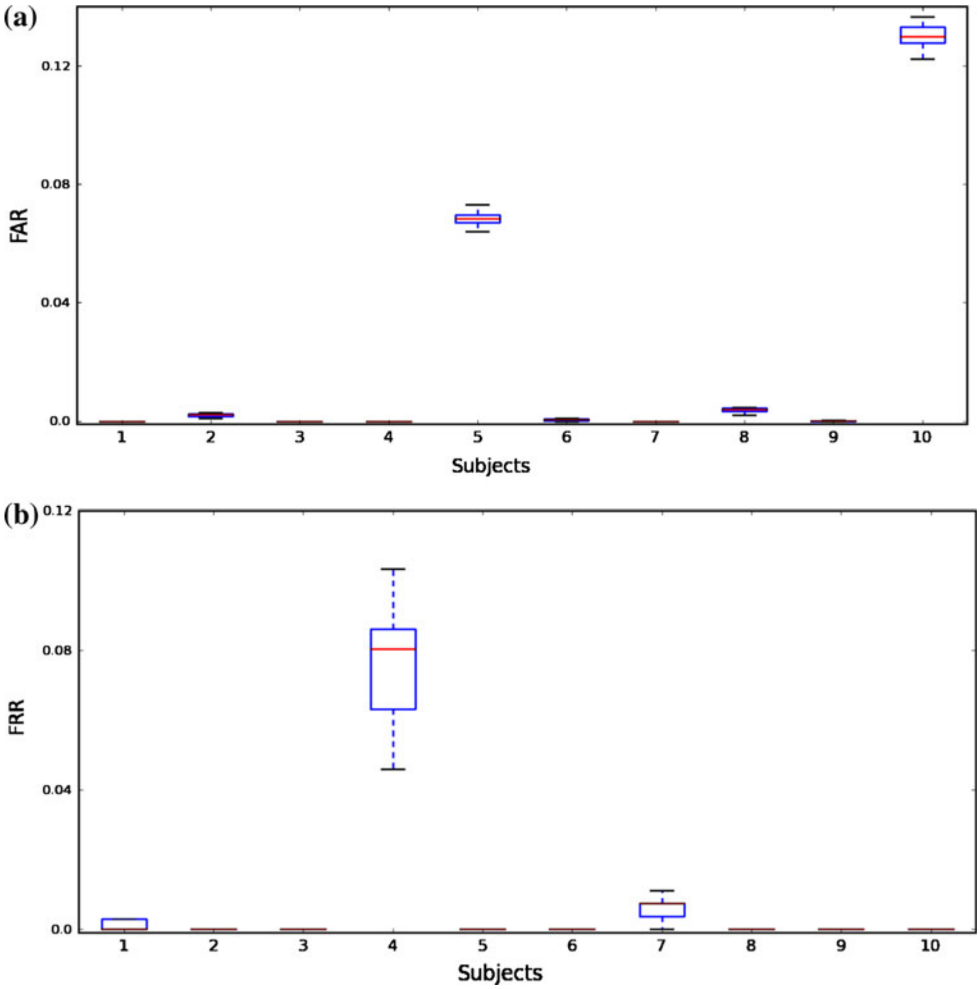


Fig. 7 Layer-4 convex hull results using temporal coherence frames 55 s in a wild scenario. **a** False acceptance rate and **b** false rejection rate

4 Discussion

In this section, we discuss several important issues concerning the results obtained.

4.1 Concerning the discriminability of the walking activity

In Fig. 9, acceleration data related to five everyday life activities are shown. Activities refer to walking, climbing stairs, standing, working seated and interacting with the environment. In the figure, the pattern arising from a walking activity is clear. Climbing stairs shows a similar pattern to walking. However, the walking regularity pattern is not present, even if some common components between the two activities can be noted. The rest of activities differ significantly from the previous ones specially in the waveform and the acceleration intensities involved. Small differences in the variation of the acceleration can discriminate between the rest of the three activities.

In Fig. 10a, walking activities for five different users are shown.

The mean value of the time series depends on the position of the sensor and it represents the rotation of the accelerometer around the correspondent axis. The pattern related to the walking activity is clear, but its shape depends on the subject. This pattern is representative of the walking-cycle, i.e., the step that a person performs after moving both legs. Systems like pedometers or step-counters are based on the information provided by this walking-cycle. The walking-cycle duration depends on the velocity of walking. Inside the cycle, the swinging movement characterizes the difference between subjects. However, in some cases, it seems difficult to find a walking-cycle, as for instance, in subject 4. In every case, the shape of the time series and its variations seem to be characteristic of the subject. This observation is empirically corroborated by the results in Table 4. Moreover, these results seem to suggest that the rest of activities display a less characterizing pattern for the user

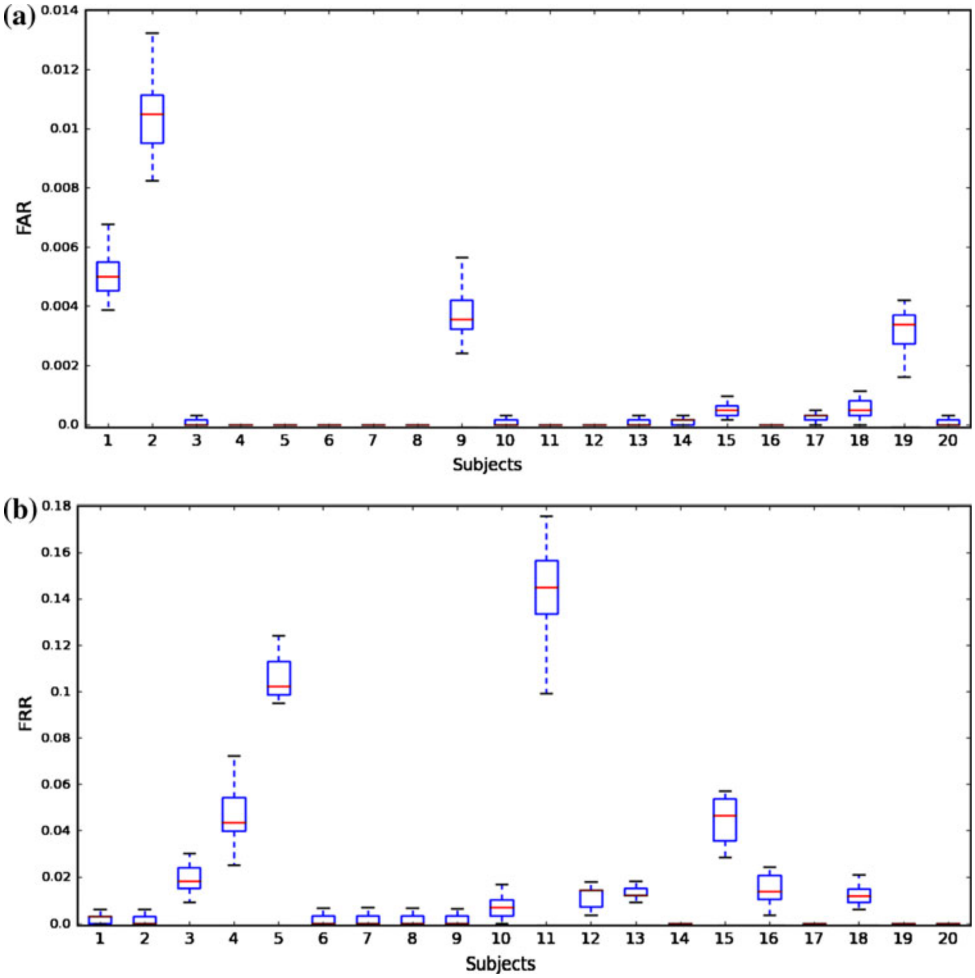


Fig. 8 Layer-4 convex hull results using temporal coherence frames 55 s in a wild scenario. **a** False acceptance rate and **b** false rejection rate

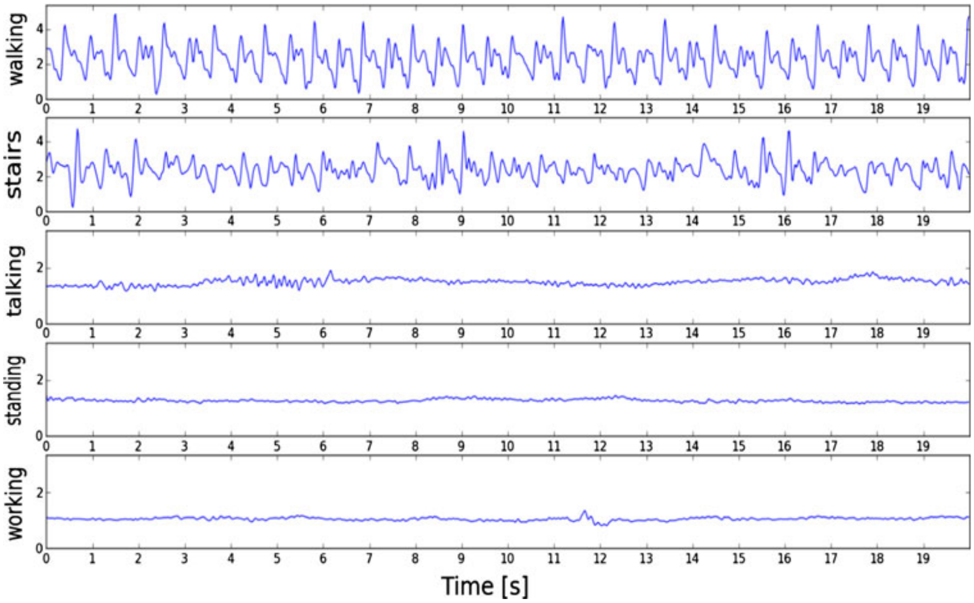


Fig. 9 Accelerometer data of five activities

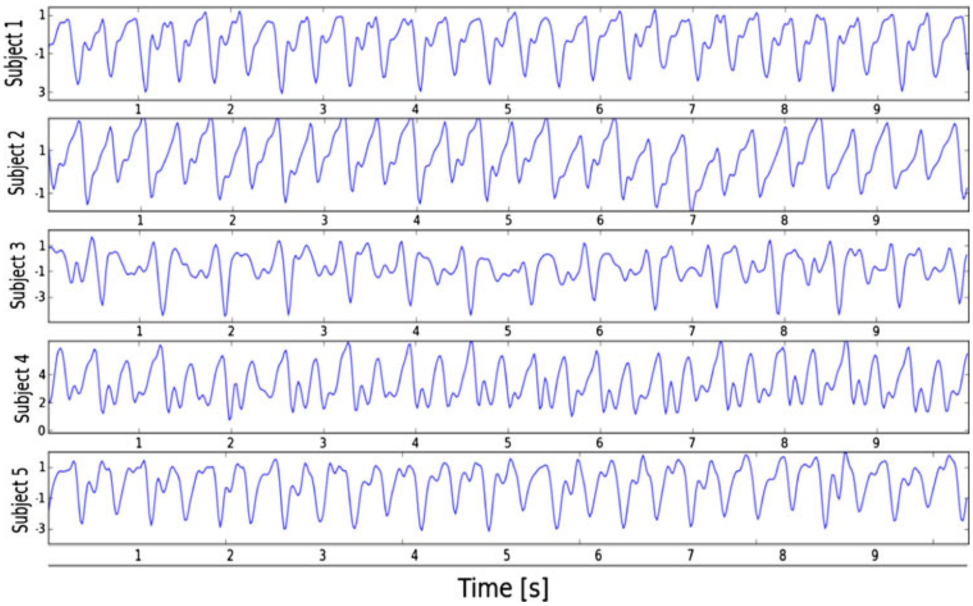


Fig. 10 Accelerometer data of walking activity for five different subjects

which is probably due to a randomness associated with the activity definition.

4.2 With respect to personalization in the general activity recognition subsystem

After the personalization of the general activity recognition, the system is much more selective with respect to the new walking pattern. This effect reduces the performance of the activity classifier for the rest of the subjects and filters out part of the walking patterns from other users while keeping the discrimination performance for the desired user. At first glance, this effect could be surprising since one expects the performance for the verified user to increase. However, observe that the classification ratios are very high. Thus, even if the system further improves its verification rate the performance difference would still be small. By reducing the performance for all users except for the authorized one, this performance difference increases. This effect filters out non-walking patterns for all users. The reduction in precision for walking patterns of non-authorized users makes the system to filter many of them out. And the reduction in recall makes the system accept more non-walking patterns as walking ones for the rest of users. Hopefully and effectively, the second verification stage, which is finely tuned to user walking pattern, rejects these last ones.

4.3 Concerning users walking variability and the effect of the layer-3 convex hull algorithm

Most of state-of-the-art methods use a very small number of runs from the users in the dataset when performing walking activity. Even worse, these runs are usually performed in very controlled conditions with little terrain variability. This severely under-represents users walking variability and makes the task of user verification much more simple. For this reason, in this article, we try to provide results using data acquired ‘in the wild’ such as in rough terrains or in adversarial conditions, in crowded places or with obstacles. This feature is important because the variability of the walking pattern for one user can be very high. In Fig. 11, a scatter plot of Feature 1 versus Feature 2 computed from the acceleration time series are shown. Features data points are relative to the same subject but they belong to two different walking paths. Although clusters are close, it is clear that using just one walking run is not enough for modeling users walking activity.

In our algorithm, layer-3 convex hull uses the mixture of hulls in order to better represent the user’s walking pattern. Given a data set in a temporal frame of an authorized user, if this set is not well represented, a new model—layer-2 convex hull—is added to the user walking profile. Figure 12 shows the effect of adding up to six models to the user’s profile on the false acceptance and false rejection ratios. These empirical results reinforce the intuition and claim stated in the former lines about the unsuitability of using one or two runs for verification purposes.

Observe that FRR is very high while the FAR is nearly zero at the first step. This confirms that the first model does not represent very well the diversity of the walking data of one user. As new data are observed and new models are added, the FRR drastically reduces and FAR is barely affected. The final results achieved after the construction of the layer-3 one-class verification system are average FAR is 0.01604 and the average FRR is 0.3.

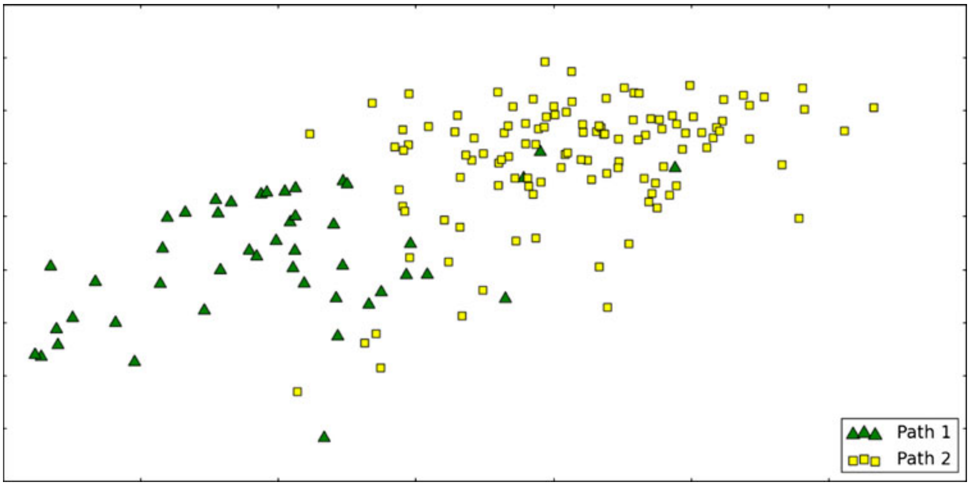


Fig. 11 Scatter plot of feature 1 versus feature 2 showing separation between walking runs in two different paths on the same subject

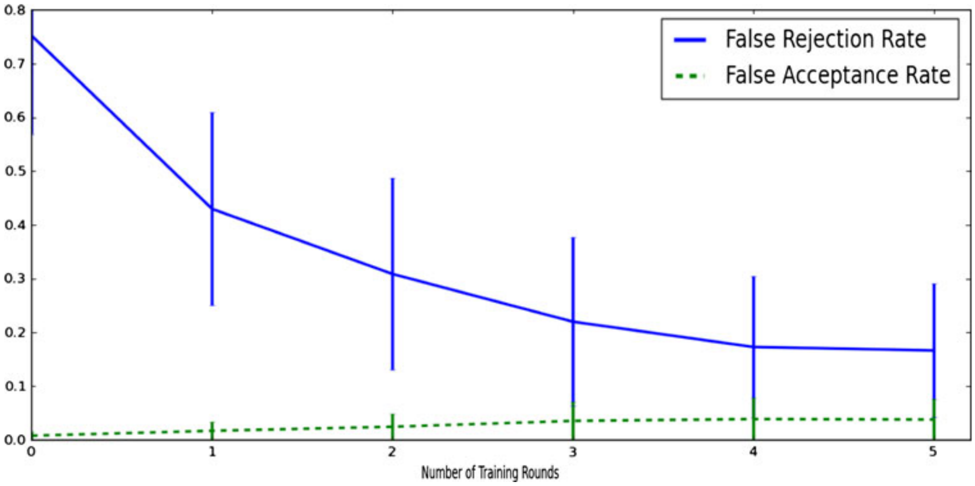


Fig. 12 Average FAR and FRR over all users when building a layer-3 hull using a commercial device

If we observe the distribution of FAR and FRR over all users in Fig. 13, one observes that the worst-case scenario for FAR is below 0.15 with six users over 0.05. If we consider FRR, the majority of the results are below 0.2.

The former results were reported on an adversarial scenario and acquisition conditions. If we consider the case of the custom wearable device using a dedicated sensor working on high resolution with data adequately sampled, the same results change considerably. In Fig. 14a, b, FAR and FRR obtained with a layer-3 convex hull algorithm with the custom wearable device are reported. Observe that in this in-vitro scenario, the system rejects a legitimate user with value 0.006 ± 0.006 and accepts a non-authorized user with 0.058 ± 0.048.

4.4 Temporal coherence and the effect of the layer-4 convex hull algorithm

The final result of the verification system is greatly improved if one takes into account temporal coherence of the data sequence. Figure 15 shows FAR and FRR as time window increases. The value in the abscissa shows how many consecutive examples of the sequence are used in the majority voting ensemble.

Using the Android-based system, 0.0001 of FAR is reached just after 55 s and 0.003 of FRR is reached after 150 s. This fact means that, in less than 1 min, no intruders are allowed in the system and, in less than 2 min, the system has a very low probability to be wrong about the authorized user. Observe that both FAR and FRR are greatly reduced as more examples are taken into account in both systems. If we consider the results reported in the former section before the time ensemble (Fig. 13) and the final results in Fig. 8, we observe a decrement of an order of magnitude in FAR and FRR in the results 'in the wild' and a reduction of half of those values in the custom scenario.

4.5 Discussion concerning the state-of-the-art verification strategies

In the following lines, we summarize the results and conditions for the most relevant state-of-the-art works. In

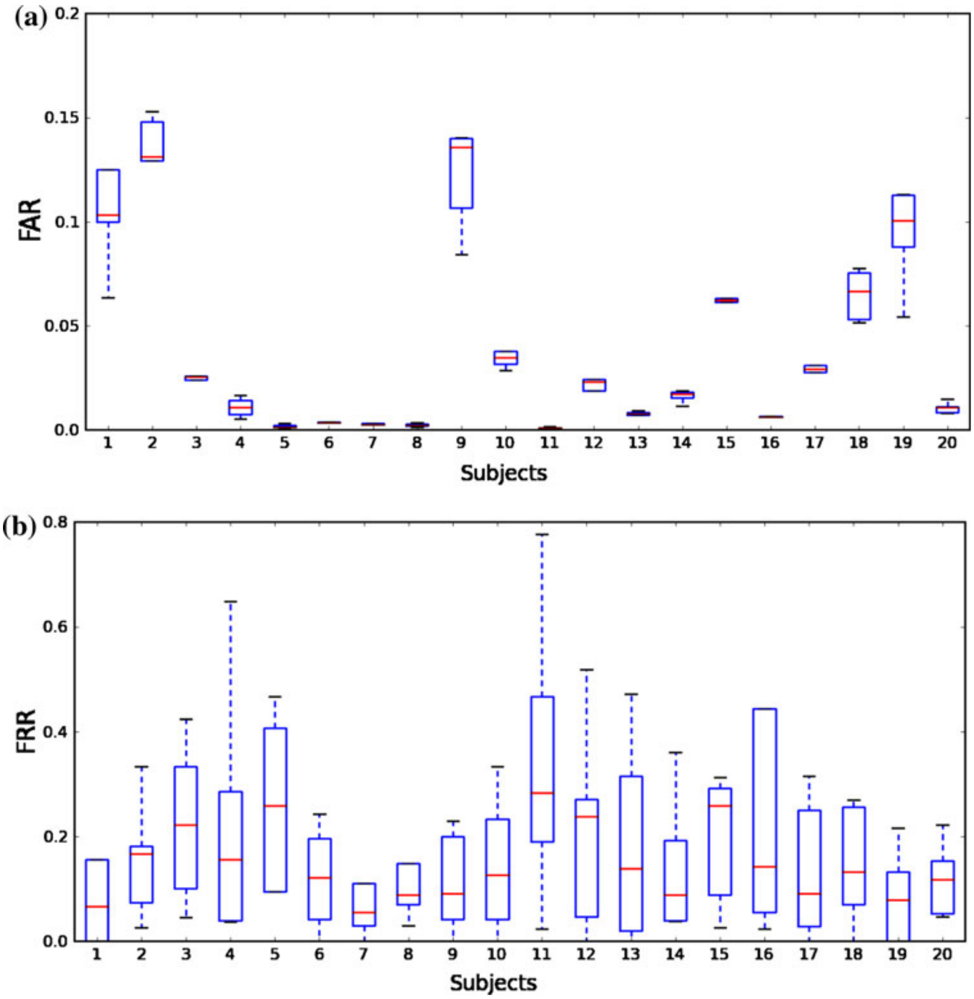


Fig. 13 **a** False acceptance ratio, **b** false rejection ratio for the layer-3 convex hull

Vildjiounaite et al. [17], experiments with voice, gait and fingerprint data have shown that in most cases FAR is around 0.01 and FRR is around 0.03. However, these results are the composite of three verification systems. If we focus on the accelerometer verification, they report an EER of 0.137. Note also that the data set is created walking along a 20 m corridor. In Mäntyjärvi et al. [11], 36 test subject for recognition walked with fast, normal and slow walking speeds in two sessions wearing the accelerometer device on their belt, at back. The best equal error rate EER = 0.07 was achieved with a signal correlation method on two runs of 20 m per walking speed. Derawi et al. [5] used a commercially available mobile device. The system was evaluated having 51 volunteers and resulted in an equal error rate of 0.20 with the system attached to the belt with two runs for each user in in vitro conditions (about 37 m down the hall on flat carpet). Finally, Gafurov et al. [8] attached the wearable system to the hip of 22 subjects performing six rounds of walking at normal speed on a flat floor. They report EER of 0.16.

Observing the former works, the general state-of-the-art verification rate is around 0.1 of equal error rate, which means that a decrement on one of the two parameters, FAR or FRR, worsens the value for the other. Observe that using our custom system, the reported values for FRR and FAR are 0.007 and 0.03 which is far smaller than the best error rates reported in literature. Note that the EER is between those values. However, EER is simple to obtain if the system is parameterizable with just one parameter. In our case, different parameters would results in many different decision error trade-off curves. Thus, we choose to report the most honest results from the cross-validation tuning of the parameters. Using the Android-based wearable device in the 'wild' scenario, we achieve values of FAR and FRR of 0.0001 and 0.003, respectively. Again, in this scenario, the results are far better than the best reported, but in a hard and adversarial scenario with several obstacles.

5 Conclusions

In this article, a novel personalized technique for user authentication and verification using gait as a biometric unobtrusive pattern is proposed. The method is based on a

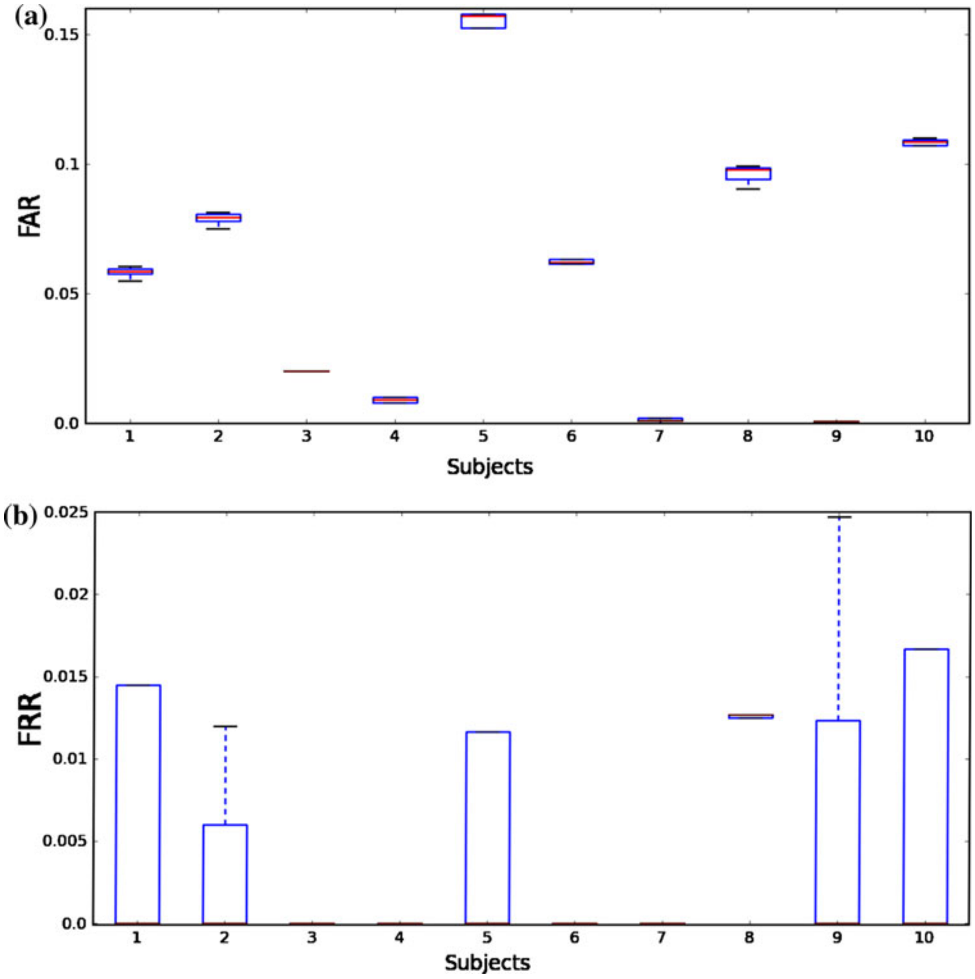


Fig. 14 Results obtained using a layer-3 verification subsystem: **a** false acceptance rate, **b** false rejection rate

two stages pipeline. First, an activity recognition classifier is able to distinguish the walking pattern of any user with respect to four other usual activities. This classifier is personalized for a specific user using a small sample of her/his walking pattern. As a result, the system is much more selective with respect to the new walking pattern. This effect reduces the performance of the activity classifier for the rest of the subjects. This effectively filters out part of the walking patterns from other users while keeping the discrimination performance for the desired user.

A second stage verifies whether the user is an authorized user or not. This stage is defined as a one-class classification problem. In order to solve this problem, a four-layer architecture is built around the geometric concept of convex hull. Each layer covers different needs. The first layer concerns computational complexity and storage requirements. The second layer improves convex hull performance by adding robustness to noise and outliers by means of a bagging ensemble procedure. The third layer allows the use of multiple convex hulls for modeling non-convex shapes. Finally, the fourth layer takes into account temporal coherence to boost the results of the verification system.

Two different scenarios are proposed as validation with two different wearable systems. A custom high-performance wearable system is built and used in a free environment. Using this system, a data set of ten users is gathered during several days, recording five usual activities: walking, climbing stairs, standing, interacting with the environment and working seated. A second dataset is acquired from an Android-based commercial device. In this last experiment, twenty subjects freely perform seven runs in a ‘wild’ scenario with rough terrains, adversarial conditions, in crowded places and with obstacles.

Results on both systems and datasets are FRR $= 0.0072$ and FAR $= 0.03$ for the custom system, and FRR $= 0.003$ and FAR $= 0.0001$ for the commercial system. These results are very encouraging and, to the best of our knowledge, improve the verification rates from gait patterns compared with state-of-the-art techniques.

All the results are obtained setting the accelerometer sensor in the upper torso of a person. In near future, we plan on extending the system for handling data acquired from the sensor located at different parts of the body. Another interesting problem arises when the number of

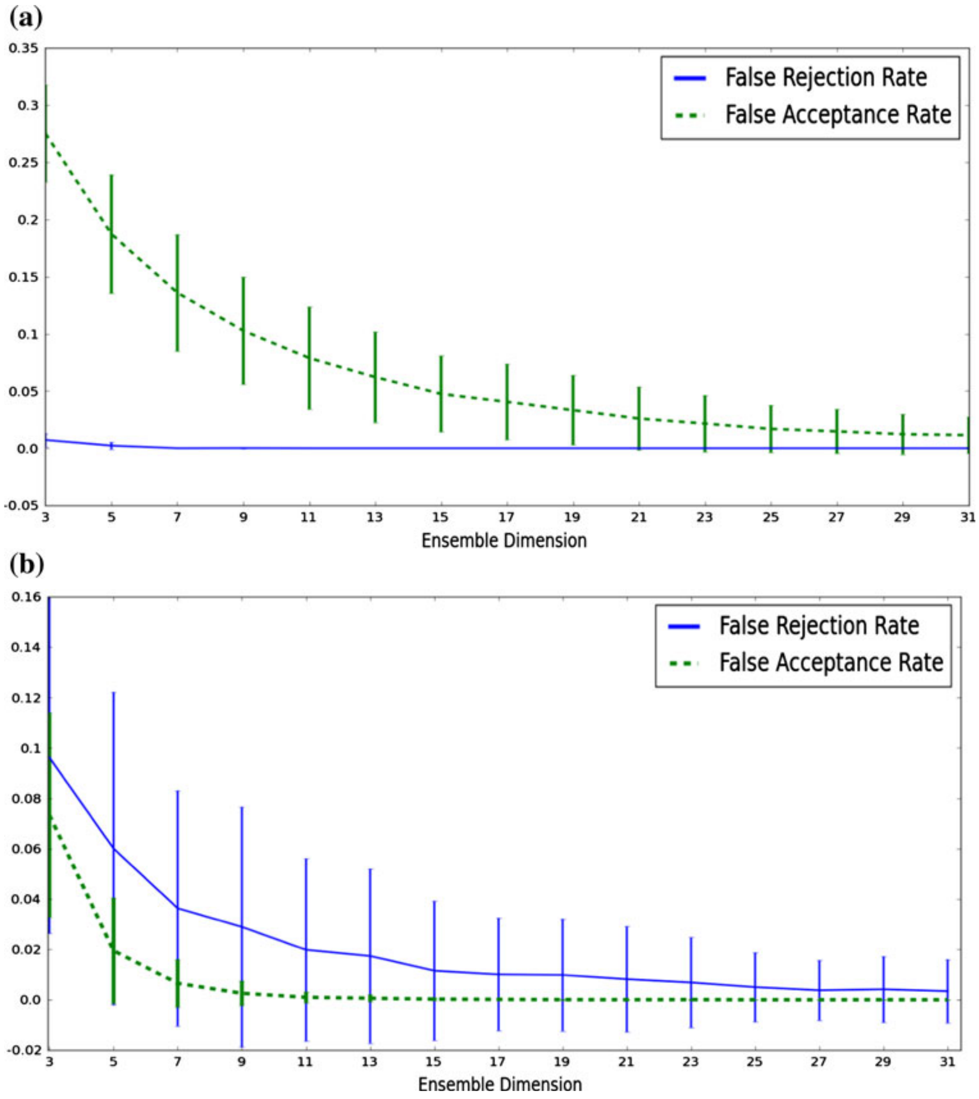


Fig. 15 FAR and FRR evolution as the temporal ensemble size increase in **a** the custom wearable system and **b** in a commercial device

authorized users increase in the system. A general verification system considering just authorized users patterns is not feasible, since FAR will undoubtedly increase due to the fact that there is more feature space considered as authorized. An effective way of handling this problem is to create and automatically select user profiles.

Acknowledgments This work is partially supported by a research grant from projects TIN2009-14404-C02, La Marato de TV3 082131 and CONSOLIDER (CSD2007-00018).

References

1. Bao L, Intille SS (2004) Activity recognition from user-annotated acceleration data. In: Pervasive. Springer, Berlin, pp 1–17
2. Bennett KP, Bredensteiner EJ (2000) Duality and geometry in svm classifiers. In: Proceedings of the seventeenth international conference on machine learning, ICML ’00. Morgan Kaufmann Publishers Inc, San Francisco, pp 57–64
3. Bianchi L, Angelini D, Lacquaniti F (1998) Individual characteristics of human walking mechanics. Pflügers Archiv Eur J Physiol 436:343–356
4. Breiman L (1996) Bagging predictors. Mach Learn 24:123–140
5. Derawi MO, Nickel C, Bours P, Busch C (2010) Unobtrusive user-authentication on mobile phones using biometric gait recognition
6. Dietterich TG, Bakiri G (1991) Error-correcting output codes: a general method for improving multiclass inductive learning programs. In: Proceedings of AAAI-91, AAAI Press, Anaheim, pp 572–577
7. Freund Y, Schapire RE (1999) A short introduction to boosting
8. Gafurov D, Snekkenes E, Buvarp T (2006) Robustness of biometric gait authentication against impersonation attack. In: Meersman R, Tari Z, Herrero P (eds) On the move to meaningful internet systems 2006: OTM 2006 workshops, lecture notes in computer science, vol 4277. Springer, Berlin/Heidelberg, pp 479–488
9. Lester J, Choudhury T, Borriello G (2006) A practical approach to recognizing physical activities. In: Proceedings of pervasive, pp 1–16
10. Mannini A, Sabatini AM (2010) Machine learning methods for classifying human physical activity from on-body accelerometers. Sensors 10(2):1154–1175

11. Mäntyjärvi J, Lindholm M, Vildjiounaite E, marja Mäkelä S, Ailisto H (2005) Identifying users of portable devices from gait pattern with accelerometers. In: IEEE international conference on acoustics, speech, and signal processing
12. Preparata FP, Shamos MI (1985) Computational geometry: an introduction. Springer, New York
13. Ravi N, Nikhil D, Mysore P, Littman ML (2005) Activity recognition from accelerometer data. In: Proceedings of the seventeenth conference on innovative applications of artificial intelligence (IAAI). AAAI Press, California, pp 1541–1546
14. Sprott JC (2003) Chaos and time-series analysis. Oxford University Press, Oxford
15. Tax DMJ, Juszczak P (2002) Kernel whitening for one-class classification
16. Terrier P, Schutz Y (2003) Variability of gait patterns during unconstrained walking assessed by satellite positioning (GPS). Eur J Appl Physiol 90(5–6):554–561
17. Vildjiounaite E, Makela SM, Lindholm M, Kyllonen V, Ailisto H (2007) Increasing security of mobile devices by decreasing user effort in verification. IEEE Comput Soc 80