

# Biometric walk recognizer

## Gait recognition by a single smartphone accelerometer

Maria De Marsico<sup>1</sup> · Alessio Mecca<sup>1</sup>

Received: 14 November 2015 / Revised: 29 March 2016 / Accepted: 1 June 2016 /  
Published online: 28 June 2016  
© Springer Science+Business Media New York 2016

**Abstract** This paper presents an approach to gait recognition based on a single consumer accelerometer, built in most present mobile devices. It does not propose a completely novel algorithm, but rather investigates better ways to exploit the Dynamic Time Warping (DTW), which is still one of the most used at present in literature. To this aim, the paper presents both a new segmentation algorithm to split the gait signal into cycles/steps, and investigates the best way to use the possibly segmented signal for recognition. Summarizing, the first contribution of the present work is the proposal of a new segmentation algorithm for the gait signal, which does not require any pre-processing, either interpolation or noise reduction, to enhance the original signal, and its comparison with two other state-of-the-art step segmentation algorithms. The second contribution is related to the extensive tests performed with the five different investigated matching methods. The tests are carried out exploiting all compared segmentation algorithms and three different datasets, collected using different sensors: the originally exploited BWR dataset, that includes walk templates from 30 volunteers, and two huge datasets used for this kind of testing, namely the ZJU-gaitacc and the OU-ISIR Inertial Sensor Database. Tests have been performed in both verification mode, either single-template or multiple-template, and identification mode, both closed and open set. The latter is rarely found in literature though representing the most frequently predictable applicative setting. It is worth underlining that the final goal is to allow using low-cost, built-in sensors that nowadays equip most smartphones. The best result in closed set identification, which is the identification mode usually reported in literature, is achieved using the most constrained method, i.e., limiting the walks in the gallery and in the probe to have a similar number of steps. It reaches  $\approx 93\%$  of Recognition Rate (RR) on ZJU-gaitacc dataset. The best result obtained with methods exploiting segmentation to overcome

---

✉ Alessio Mecca  
mecca@di.uniroma1.it

Maria De Marsico  
demarsico@di.uniroma1.it

<sup>1</sup> Sapienza University of Rome, Rome, Italy

the mentioned limitation reaches  $\approx 83\%$  of Recognition Rate (RR) on the same dataset, using our proposed algorithm. The best results in verification is achieved using multiple templates per user, again without segmentation, with an Equal Error Rate (EER) of 0.09, while the best results with segmentation is achieved again with our algorithm and is an EER of 0.10. This is a very good result for a soft biometrics as gait if often considered. As expected, open set identification achieves lower performance.

**Keywords** Biometrics · Gait recognition · Accelerometer · Mobile devices

## 1 Introduction

Biometrics can provide identity recognition as well as authentication for many applications, from ambient intelligence to the control of access to critical and protected zones/services. Some biometrics, mostly physical traits like, e.g., face, iris, and fingerprints, can be used to uniquely identify a person, though often requiring a controlled setting. They are usually classified as "strong". Other traits can lack this ability because they rather identify a class of users, e.g., gender, or can be less reliable, for example because they can be affected by several factors, e.g., environmental or emotional conditions, like most behavioral traits. For these reasons they are usually classified as "soft" biometrics. Even soft biometrics can be exploited for individual recognition, if used in combination with other soft biometrics or as an enforcement of strong ones, according to multibiometric protocols. Gait recognition, i.e. recognizing people from the way they walk, is often considered as belonging to the latter category. At present, gait recognition is mostly adopted in literature in verification modality, to just recognize the owner of a specific device, and at present no real scenario with massive gait recognition can be identified. In other words, the really interesting applications are usually bound to a white list modality, which usually entails a number of enrolled users which is not significantly high.

Gait recognition techniques can be classified into three different categories based on the technological settings investigated so far:

- machine vision-based: these methods use one or more cameras for the (video) data acquisition and suffer from typical image processing issues, e.g., occlusions and illumination variations;
- floor sensor-based: they require specific ambient equipment, therefore cannot be used everywhere and require a preliminary complex set-up;
- wearable sensor-based: they generally use accelerometer as well as other kinds of wearable sensors for data acquisition.

Given the limitations of the first two groups of techniques in terms of feasibility and availability, our choice is to focus on the third group, and in Section 2 we discuss more in detail this kind of systems, mentioning works that use a single accelerometer, more than one accelerometer or the combination of accelerometer and gyroscope. The goal of the work in this paper is to exploit the simplest possible set-up, using off-the-shelf, widespread equipment, and possibly only one sensor. In other words, the aim is allowing users to undergo this authentication/recognition modality without need of special equipment, and without carrying out any specific action. For this reason, the proposed system captures gait features using currently available medium profile smartphones. Such devices generally have sufficient quality accelerometers on-board. The gyroscope, even if embedded, is not used, in

order to maintain a better compatibility across different mobile devices. In fact, a possible disadvantage of using gyroscope too, it that at present it is not a standard component for most mobile devices, especially taking into account also the less recent models. Moreover, the use of the gyroscope would both add further complexity to the system architecture and increase the computational cost of the matching. The possible advantages would obviously regard an increased recognition accuracy; however there is no general agreement on this point: even according to the results reported in [11] it seems that the use of gyroscopes in connection with accelerometers may even decrease performance in terms of EER.

This paper extends the work presented in [1]. As most methods in literature, the developed system uses Dynamic Time Warping (DTW) to separately match the signals over the three accelerometer axes, and then linearly combines the results. The main goal is to investigate how to get better performance from a basic implementation of DTW. As a first contribution, the new step/cycle segmentation procedure proposed in [1], is better detailed and results obtained by using it prior to matching are compared with two other ones in state-of-the-art. Huge pre-processing aiming at enhancing the signal quality (smoothing and noise reduction, interpolation, and time normalization) is deliberately avoided to better explore the potentialities of DTW in itself. Given such choice, it was not possible to reliably compare our algorithm with further ones where segmentation is more deeply dependent on preliminary operations on the signal. It is worth noticing that recognition by a single accelerometer achieves fair performance. One of the goals of a lightweight processing is to allow the recognition either locally, e.g., on the smartphone itself, or remotely on a computer receiving gait data, to be possibly fused with vision-based data from cameras.

Based on segmentation results, [1] proposed different strategies to improve DTW flexibility, represented by different matching algorithms. A comparison between them was carried out in order to evaluate the best compromise between accuracy and acquisition constraints. We consider that recognition performance can be measured according to different modalities. In verification, the user claims an identity and a 1:1 matching with such identity is carried out. Most approaches in literature use (implicit) verification, since the identity claim is implicit in the preliminary enrollment of a single user, who is the owner of the device. These algorithms aim at verifying if the user keeping the device is its true owner. In closed set identification, no identity is claimed by the user, therefore a 1:N matching must be carried out against all registered users, and the probe is always assumed to belongs to a subject in the system gallery. In open set identification, no identity is claimed by the user (1:N matching) and probe user may not be known to the system, therefore a threshold-regulated reject option is added. The investigated techniques were tested in [1] in closed set identification and verification modalities, while in the present paper the full set of modalities is tested. Even if it is possible to use either a distance or a similarity metric as matching result, is is always possible to convert one into another, and given that we use a distance, such kind of measure will always be mentioned. It is worth pointing out some further differences among the described modalities. In closed set identification, the only requirement is that the right subject appears in the first position of the result list, notwithstanding the distance between the probe and the returned template. In verification modality, a threshold constrains the acceptance of the matching as a genuine one. In open set identification mode, a distance threshold regulates the acceptance of a template as in verification: the degree of similarity must be very high to obtain a correct answer. However, in addition, in open set identification modality, the a template belonging to the right identity must also be the first in the returned list. If a gallery template is close enough, but not the closest one to the probe, its identity is not the one (possibly erroneously) returned as the probe identity. In a similar situation, in verification modality, if the similarity is just sufficiently good, the system

might (possibly erroneously) accept the identity of this gallery template as the identity of the owner of the device. Experimental results in the previous paper refer to a dataset collected using the accelerometer built-in in the OnePlus One smartphone, and on ZJU-gaitacc, while in the present work also presents results achieved on OU-ISIR dataset. It is to point out that at present the latter two are among the largest public datasets collecting inertial sensors data available for gait recognition. Recognition can be carried out on an external server, and this allows combination with other biometrics to secure the access not only to personal devices but also to critical locations or services. The paper continues as follows. Section 2 discusses some proposals dealing with sensor-based gait recognition. Section 3 sketches the overall framework that we plan to develop, and the relevant aspects of the already implemented recognition procedures. This section also includes a detailed description of the proposed segmentation algorithm. Section 4 presents the experiments setup. In particular, it describes the two segmentation algorithms that we compared with our one and describes in more detail the datasets used in the tests to compare the different recognition approaches. Section 5 shows the results achieved and the comparison between the different techniques. Finally, Section 6 draws some conclusions and outlines the main steps of our future work.

## 2 Related work

During the last years the approaches based on wearable sensors have increased their popularity over computer vision-based and floor sensors-based ones. This is due to three main factors. The first one is the low cost of these sensors, in particular accelerometers, which are the most suited ones for this task, and their widening spread on mobile devices. The second factor is the acceptable computational cost of the recognition algorithms generally used. Of course, last but not least, also the good results that can be achieved contribute to the interest for these approaches. Just think that nowadays every mobile phone has at least one built-in accelerometer and a lot of them have gyroscope and magnetometer too. For these reasons, it is possible to create low cost recognition systems that can be used both directly on the smartphone or tablet, generally for verification (e.g., recognition of the owner), and on a remote computer, generally for identification (e.g., access grant to a critical zone). Moreover, differently from the other two categories of approaches, a built-in accelerometer can follow the user everywhere without blind spots or other scene distortions (affecting computer vision techniques) and without need to modify the environment (requested by equipped floors).

A comprehensive survey of all the three kinds of approaches to gait recognition can be found in [4] while [9] provides a review of machine vision-based systems, that represent anyway the most used ones so far. This section will only mention some of the wearable sensor-based recognition system which are closer either in algorithmic setting or in aim with the proposal in this paper.

The first general remark is that, in the majority of papers on accelerometer-based gait recognition, the purpose is the recognition of the owner of the phone. Therefore, these systems work in verification modality. Performances are generally estimated using an all-against-all matching in order to derive overall False Acceptance Rate (FAR), False Reject Rate (FRR) and Equal Error Rate (EER), achieved by the acceptance threshold where  $FAR=FRR$ . The work proposed here mainly carries out identification in both closed set and open set modalities. In the first case, performance is estimated by Recognition Rate (RR) and Cumulative Match Curve (CMC); in the second case, Detection and Identification Rate

(DIR) and EER are used instead. However, the proposed system can be used in verification modality too.

It is worth pointing out that most papers rely on identifying cycles in the walk signal, intended as a pair consisting of a right and the following left step or vice-versa. The segmentation algorithm in Section 3.2 rather detects single steps.

In [3], authors present a system that uses a low sampling rate accelerometer, the one built in Google G1 phone. The system achieves an ERR of 20 % using a dataset collected from 51 subjects. The acquisition procedure requires a straight way walk in a 37m hallway, 2s of stop and return. The recognition procedure uses cycle detection, computes the average cycle and then applies Dynamic Time Warping (DTW) algorithm for matching. In [2], the same authors use a better accelerometer (the Motion Recording 100), improve the preprocessing phase with a noise reduction algorithm, and perform an outlier steps detection and removal. The latter allows disregarding steps that are too different from the others, possibly due to noise or very short temporary variations. The performance of the system increases to an EER of 5.7 %.

The works cited from now on use accelerometers with similar quality of the one used in [2]. This is because nowadays most accelerometers that are built in smartphones have a similar sampling rate of about 100 samples/second.

The work in [5] deals with a system that uses two different approaches to recognition. The first one uses similarity between histograms computed from the gait signals and achieves a 5 % of EER. The second one uses cycle group comparison and achieves a 9 % of EER. These results are obtained using a smaller dataset (21 users with walks collected by "AVR Butterfly Accelerometer - Motion Recording 100" in a 70m hallway) compared to the others.

The authors of [8] present a more comprehensive framework for gait recognition. The system gets raw data from both the accelerometer and gyroscope built in an Android smartphone not better identified. The framework uses continuous wavelet transform time frequency spectrogram analysis for feature extraction, and cyclostationarity analysis for matching. The dataset collects walk signals from 36 users. The smartphone is positioned in the right pocket with the phone facing outwards and oriented vertically. The walk is straight down a hallway (about 25m long) and return. Results report performance at different levels of walk speed. The tests achieve a 99.4 % of verification rate at 0.1 % of FAR for pace walk vs. pace walk, 96.8 % for fast walk vs. fast walk, and a 61.1 % for pace walk vs. fast walk. Achieved results are very good but this system uses the gyroscope too, differently from the others.

In [13] the authors use the Hidden Markov Model (HMM) technique. The system proposed achieves a 10.42 % of False Non Match Rate and a 10.29 % of False Match Rate with a dataset of 48 users following the same acquisition protocol of [3]. A HMM is trained for each user using his/her unsegmented walks (so 48 HMMs are created) and all the registrations for all users are submitted in turn as probes.

None of the above works reports results for identification. Moreover the exploited datasets are not shared by other works in literature.

The paper in [14] presents a high performance system, using five accelerometers attached in various parts of the body. The authors carry out identification by a method based on signature points. Also this method does not use the cycle division. It achieves a RR of 96.7 % with a dataset of 30 users. The work in [19] is an evolution of this system that can be used in for identification and verification modalities. The recognition method uses signature points as before, but applies a preliminary clustering phase to group these points, with a relevant improvement of the performances. The authors also created a huge dataset

(with walk templates collected from 175 different subjects) of accelerometer data, which is publicly available for non-commercial use<sup>1</sup>. This dataset was used for our experiments, therefore further details on its composition will be given in Section 4.1.2. The tests on this dataset produce a RR of 95.7 % for identification and a 2.2 % of EER for verification. This method achieves the best results among the mentioned ones, but it is worth noticing the use of a very complex equipment for data acquisition, not so suitable in a real setting.

A further publicly available dataset is presented in [11]. The authors propose a valuable work in this field and make available to the research community one of the largest inertial sensor datasets for gait recognition purposes, namely OU-ISIR Inertial Sensor Dataset (from now on OU-ISIR). Since also this dataset was used in our experiments, further details are reported in Section 4.1.3. The authors also show the performances of their cycle detection method and the comparison with some other techniques in the state-of-the art, namely the ones proposed in [2, 6, 15, 18]. Actually, as also stated by the authors, it is difficult to directly compare the performances by the original databases with those achieved on the new database, due to the large difference in the number of subjects and to the very different acquisition conditions, mostly the kind of sensors and their positioning that heavily affect the acquired signal. However it is interesting to notice that the achieved EER increases from about 6 % for almost all methods with the original datasets, to a 14-20 % EER with the new dataset. The section reporting experimental results will further discuss aspects related to performance decrease when using this dataset.

Another interesting work, proposed by the same authors of OU-ISIR dataset, can be found in [12]. Its aim is to normalize acceleration data in order to address the problem of a different orientation of the data acquisition sensor/s. The paper assesses the advantages of using this kind of preprocessing, and the significant increase in term of performances, which are reported for verification modality only. The proposed preprocessing can be very useful in all the situations where the sensor/s have a wide possibility of changing their position. A different approach to the same problem can be found in [20].

A wider review of gait recognition based on inertial sensors is out of the scope of this paper, but interested readers can refer to [17]. However to the best of our knowledge we can observe that no work in literature reports gait recognition results in open set identification modality, which actually is the one that corresponds to a more realistic identification scenario, where not all users presented to the system have been previously enrolled in the system.

### 3 Proposed approach

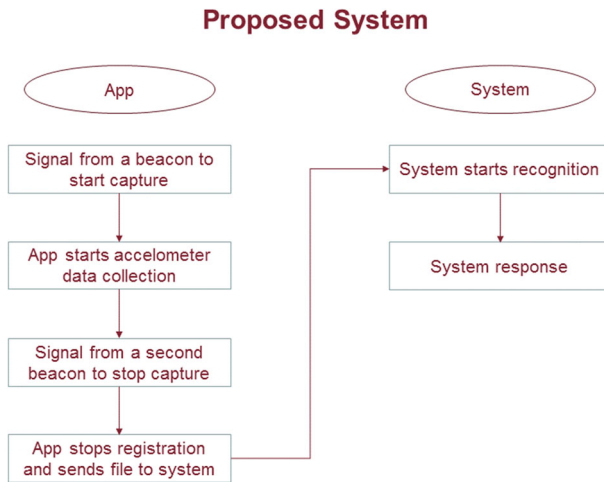
This section shows the core recognition procedures developed in this work and the overall design of a complete system for automatic user recognition.

The complete system will be composed by two different modules that will be able to automatically communicate. The first module will run on a smartphone with a built-in accelerometer for the data acquisition. The second module will run either on the phone itself (if in verification modality), or on a desktop computer (if in identification modality), to perform the recognition, as shown in Fig. 1.

The figure points out that, in case of remote transmission to a server, suitably positioned beacons, e.g., close to the entrance of a restricted area, can trigger the start and the end of

---

<sup>1</sup><http://www.ytzhang.net/datasets/zju-gaitacc>



**Fig. 1** Schema of an automatic system for gait recognition

the data capture, via communication through low-consumption Bluetooth. Beacons can be substituted by any kind of device that can remotely trigger actions through communication. At present, a dedicated app for the capture module is being developed to this aim. In the meanwhile, data collection exploits an off-the-shelf Android application, namely *Physic Toolbox Accelerometer*<sup>2</sup>, free on the Google Play Store.

The recognition module, which is the true core of the application, is already implemented for desktop processing. As for now, the two modules do not automatically exchange data because this function is not provided by the chosen app, but will be included soon in the dedicated app. It is worth underlining that, in general, if in verification modality, transmitting data to a remote computer would create useless privacy issues, therefore it is worth implementing a local recognition procedure, that will be soon added to the mobile dedicated app. On the other hand, if in identification modality, e.g., to control access to a restricted zone, it would be impractical to store and maintain the gallery of enrolled users on each mobile device, so that the best choice is to transmit a (protected) template to an authentication server. This latter option is the one already implemented, and it would also allow enforcing the recognition accuracy by fusing results with those from different sources. As an example, a computer vision-based recognizer taking as input the videos of cameras at the entrance of the zone, would send additional data as input to the "System starts recognition" activity. When applicable, using the mobile devices for data acquisition only would also significantly reduce the battery power consumption. Moreover, it is to consider that the accelerometer is the main source of smartphone energy consumption in this system: taking this into consideration its use is limited as much as possible. Actually, when communicating with a remote server, the system requires using the accelerometer just for the space between the two beacons, and, afterwards, there is only one operation of data transfer at the end

<sup>2</sup><https://play.google.com/store/apps/details?id=com.chrystianvieyra.android.physicstoolboxaccelerometer&hl=it>

of data acquisition (and not a continuous streaming). A possible argument regarding verification of the identity of an approaching user is that one could simply rely on the much simpler recognition of the mobile device, e.g., by inquiring a preset identification code. However, this would be an extremely unsafe option since the smartphone, as any other kind of physical token used for authentication, e.g., a magnetic card, can be stolen or cloned or even lost. This is just one of the issues addressed by biometric recognition.

As many researches in literature, the proposed system exploits Dynamic Time Warping (DTW) to measure the distance between either entire signals or signal segments. The reader interested to an exhaustive discussion about this classical Dynamic Programming algorithm can refer to the chapter about DTW in [10]. It is worth underlining once more that the aim of this work is not to propose a completely novel recognition algorithm but rather to investigate how to improve the performance of the most classical one. In particular, the experiments carried out aim are used to assess a novel technique for step segmentation and to investigate which is the best strategy to exploit a walk signal (unsegmented vs. segmented and, in the second case, how to match segments).

### 3.0.1 Data acquisition

For data acquisition, in the enrollment phase, a uniform protocol is applied, as usually happens in similar applications. To standardize the movements for each recording, the protocol entails the following sequence of actions:

1. to put the phone in the belt, either on the right hip or on the left one in vertical position; this grants a similar location for all users; at present the phone must be positioned with the screen facing out; the latter condition is required by the lack of an automatic triggering of start and stop of capture, so that recording is started by launching the app and by tapping on the screen both for start and stop;
2. to start the recording and then begin walking with the leg opposite to phone location;
3. to walk in straight line in the most natural way for ten steps; the number of steps can be decided in advance but all walks for enrolled users must have the same number of steps; it is to underline that this limitation only holds for enrollment, i.e., for the collection of a white list of authorized users; the experimented matching algorithms have exactly the aim to avoid it during normal system operation.

A folder is created on the desktop computer for each enrolled user, where user data are transferred (manually by now) and where results of the following processing are stored. Data for three different walks per user are acquired, after detaching the smartphone and attaching it again to the user's belt each time. However, the system can handle a different number of templates per enrolled user. While other datasets are captured by positioning the sensor always in the same place, e.g., the same side, the collected dataset is not limited by this constraint. Figure 2 shows an example of smartphone positioning.

## 3.1 Preprocessing

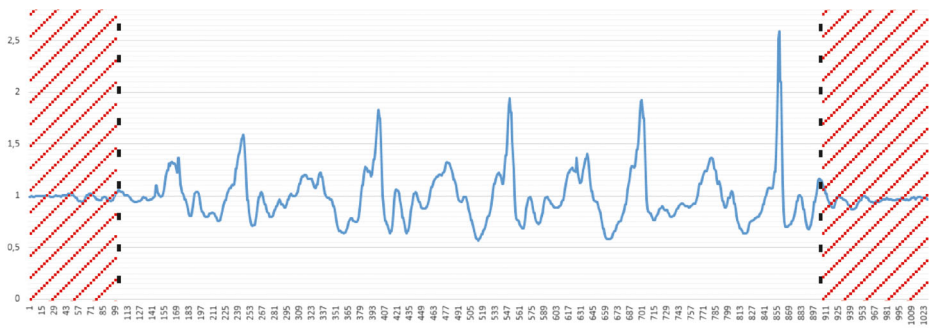
Differently from most of the systems mentioned in the related work, that perform denoising and time normalization operations, the one proposed in this paper uses only a very simple preprocessing. In order to make each stored walk signal start and stop at relevant points, a procedure identifies and discards all initial and final local maxima caused by noise. To do this, a threshold determines the set of points to discard at the extremes of the signal.





**Fig. 2** Example of smartphone positioning

This threshold is experimentally set at 1.05 on y-axis over a range of values from about -2 to +2. Once the first value greater than the threshold is found, the system searches for the first relative maximum from that point on, and discards the segment of signal before such maximum. The same procedure is carried out, starting from the end of data, to find the end of the last step. Once these points have been identified, they are saved and will be used as starting and ending points of the vector during the matching phase. This preliminary operation is carried out on the principal axis (the y-axis in our case) and then the initial and final points of the resulting portion of signal are projected onto the other two axes. Figure 3 shows an example of the discarded regions (red dashed segments) and of the resulting final signal. The threshold value is device dependent, so it cannot be applied if data came from different smartphone models. For this reason, this phase is carried out only with BWR dataset (described in great detail in Section 4.1.1). This also partially explains the different performance achieved on the database collected in-house and the public ones. To avoid this and other problems related to the differences among accelerometers, a normalization procedure of data w.r.t. the device accelerometer will soon be added to the data acquisition module.

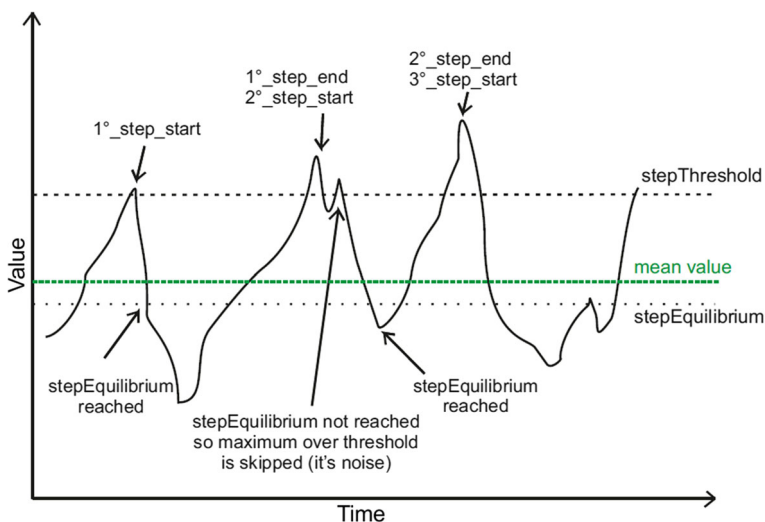


**Fig. 3** Example of extraction of the relevant segment from a signal

### 3.2 Step segmentation algorithm

This subsection presents a novel algorithm for step segmentation. The procedure works in four phases that are carried out only on the y axis. The results are then projected onto the other axes. The first and the second phases are executed only for walks that make up the gallery, and only once, namely the first time the dataset is loaded. As for probes, these two phases are substituted by a different operation, that will be detailed in the following. The third phase represents the core of the step segmentation algorithm and the fourth one is used just to improve the recognition results. These two last phases are carried out both for the walks in the gallery (only at enrollment time) and for the probe ones (according to the procedure defined below). Figure 4 shows the main parameters used by the algorithm. More details follow.

1. In the first phase, the algorithm computes the value for the variable `stepEquilibrium`, that is used to avoid an erroneous segmentation caused by noise; the value assigned to the variable is the one which is lower than the average value of the entire walk and appears with the highest frequency in the signal; `stepEquilibrium` will be used in the third phase.
2. In the second phase the algorithm finds the value for the variable `stepThreshold`; this value is obtained by taking the k-th highest relative maximum of the signal, where k is the number of steps of the walk (it is worth reminding that the system knows this value for enrolled walks because of the constraint described in Section 3.0.1); similarly to `stepEquilibrium`, this variable will be used in the third phase.
3. Figure 4 shows a graphic representation of the third phase of the step segmentation algorithm; this procedure divides the signal into steps using the variables described before and works as follows:
  - (a) look for the first relative maximum in the signal, which will be the starting point of the first step;



**Fig. 4** Example of Step Segmentation Algorithm

- (b) scan the following walk data searching for a value that is lower than `stepEquilibrium`;
- (c) look for the next relative maximum greater than `stepThreshold`, that will be the ending point of the current step and the starting point for the next one;
- (d) repeat from (b) till to the end of the entire walk signal.

If a local maximum is found after another without passing through a value lower than `stepEquilibrium`, this maximum will be discarded, classifying it as noise, as shown in Fig. 4.

4. Finally, the system looks for possible outlier steps (steps that are very different from the others in the walk); to do this, the following procedure is carried out:

- (a) compute the DTW distances between the steps identified for the walk;
- (b) for each step, compute the average distance from all the others;
- (c) compute the average of average distances, denoted as  $M$ , and the standard deviation of average distances, denoted as  $\sigma$ ;
- (d) discard the steps whose average distance from the other steps is greater than  $M + \sigma$ .

It is important noticing that the system does not discard steps whose average distance from the others is less than  $M - \sigma$ : this is because, using distances, this kind of steps are instead very good for recognition because they are in some way uniform to all the others.

As said before, the first two phases (computations of values for variables `stepEquilibrium` and `stepThreshold`) are carried out only during the collection of walks from enrolled users. In the same way, phases 3 and 4 are carried out only once for enrolled samples using the results of phase 1 and 2. In particular, the procedure performs the outlier removal for gallery samples, when requested by the adopted matching algorithm, once and for all during the enrollment phase. Also other works in literature perform outlier removal according to the same operational schema, being the differences only related to the way of choosing the steps to retain. During recognition, the probes undergo a different protocol, where phases 1 and 2 are missing. In practice, the matching operation uses an enrolled walk-fitting procedure for step segmentation, applying in turn to the probe the `stepEquilibrium` and the `stepThreshold` values computed for the gallery walk to match from time to time. Then the phases 3 and 4, the latter only when required by the adopted matching algorithm, are applied to the probe in the same way carried out for enrolled walks, before executing the chosen matching algorithm. This allows a better segmentation for data from the right user and a worst one for the others. This kind of approach, besides improving the global performances of the system, also avoids the limitation of a fixed number of steps for the probe.

To summarize, the computation of the correct values for step segmentation, that are subject-dependent, is carried out only once for enrolled users, specifically in the enrolling phase. These values are later used to compare walks from different users during testing. It is worth underlining that exactly the same segmentation procedure was carried out on both ZJU-gaitacc dataset (with satisfying results) and on OU-ISIR dataset. Given the significant difference among the signal slopes (Fig. 11 in Section 4.1.4 shows two examples of each kind of signal to better underline this point), these results are a further confirmation of the feasibility of our algorithm. As a matter of fact, differently from the walks in our dataset, there is no indication of the exact number of steps in the two external datasets, and it must be approximated. Notwithstanding this, the achieved results are acceptable, especially with ZJU-gaitacc.

### 3.3 Matching methods

For the proposed matching methods, the system uses the DTW algorithm. This procedure is often used to match two different signals that vary over time, and produces a distance value that is quite robust to misalignments, which can happen in this kind of data. In the proposed system, the DTW algorithm is executed three times in each matching, one for each axis. In order to have an effective fusion of these three results, producing a final distance score, a simple train procedure is used, when necessary, as described below in Section 3.3.7. Axes relevance depends from the device position. With smartphone in vertical position and attached on the hip, the training returns the y-axis as the most important for recognition, followed by z-axis. The resulting weights, assuming the same accelerometer orientation, are very similar to those used in other works in literature. In such position, it seems that x-axis has very little impact on the recognition. All walks achieve a very similar value on that axis, mainly because it assumes a specific meaning only in case of strong jerks or jumps, which are generally absent in natural walking.

Five different algorithms have been developed, all based on DTW procedure. The first one, WALK, is the only one that does not require the step segmentation, and as a consequence does not eliminate outliers, while the last one, STEPS SLIDING WINDOW, uses step segmentation just for alignment, but does not eliminate the outlier steps.

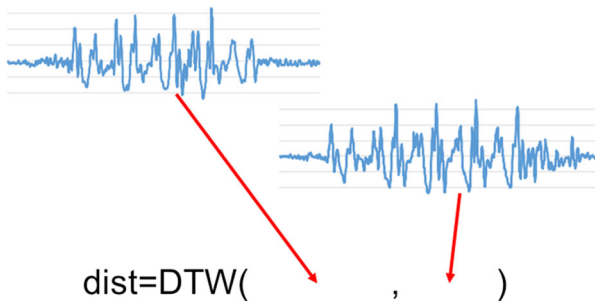
#### 3.3.1 Walk

The simplest, but also most accurate recognition method is WALK (Fig. 5).

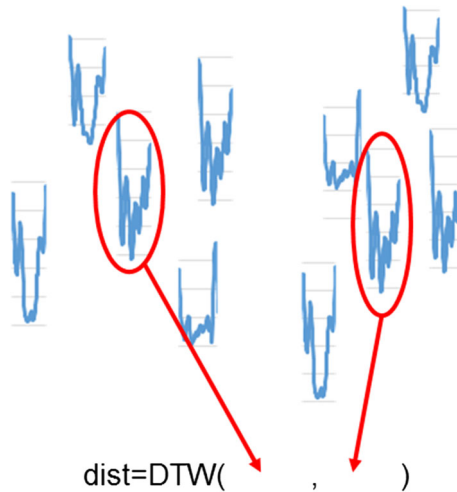
After the simple preprocessing described above to discard initial and final walk noise, it simply uses the DTW algorithm on the probe and all enrolled walks (possibly more than one for each user). As said before, it does not need to carry out step segmentation. The real problem with this algorithm is a strong constraint: the probe and the enrolled walks must have a sufficiently similar number of steps, otherwise the performances are very low. This problem, however, is not so relevant if the system acquires data always in the same points, for example between two beacons, as described in Section 3. As a matter of fact, if beacons are suitably located, it is probable that the number of steps performed by users will be fairly equal to that decided for enrolling.

#### 3.3.2 Best step

The first attempt to avoid the constraint represented by using a fixed number of steps is to compute the "best step" for each walk, that we define as the centroid of the cluster composed by its steps (Fig. 6).



**Fig. 5** Graphic example of WALK matching: after discarding initial and final noise, the entire signals are matched



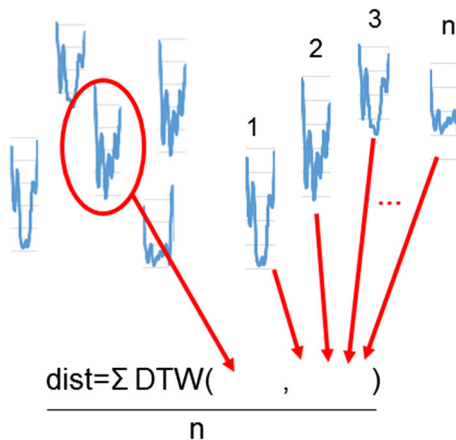
**Fig. 6** Graphic example of BEST STEP matching: the centroids of the two signals are matched

After segmentation, centroids are those steps having the minimum average distance, computed by DTW, from the others in the segmented walk. Computing this best step requires no additional computational costs: when the system computes the outlier steps it obtains the centroid step for free. For each gallery walk to match, the distance between the probe and such walk is computed by applying DTW between the best steps of the two walks.

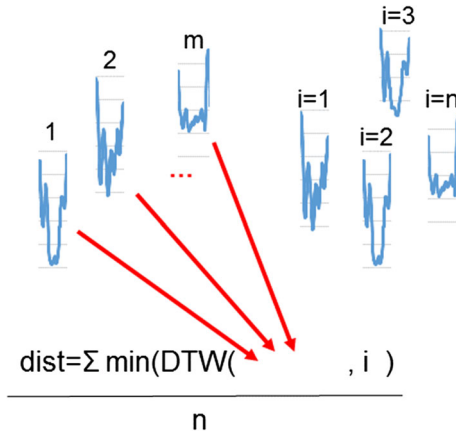
### 3.3.3 Best step vs all

This method uses the same concept of BEST STEP and is denoted as BEST STEP VS ALL (Fig. 7).

For each gallery walk to match, this procedure requires computing the average DTW distance between the best step of the gallery walk and all steps in the probe.



**Fig. 7** Graphic example of BEST STEP VS All method: the centroid of the steps of the gallery walk is matched against all each step and the average is returned



**Fig. 8** Graphic example of ALL STEPS VS ALL method: each step in the probe is matched against each step in the gallery walk, the minimum distance is selected, and the average over all minima for probe steps is returned

3.3.4 All steps vs all

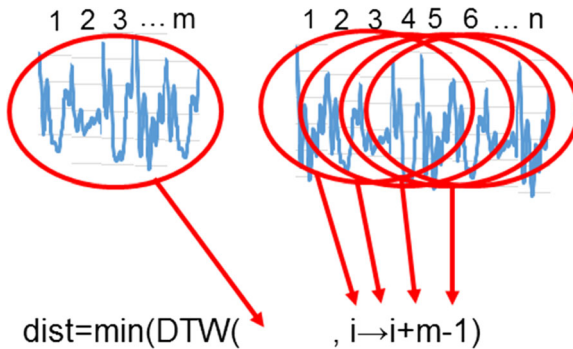
A different kind of approach has been tried with ALL STEPS VS ALL, ALL STEPS for short (Fig. 8).

After preprocessing and step segmentation, this method, for each gallery walk to match, computes the DTW distance from each step in the probe to each step in the gallery walk, and takes the minimum such distance. Then it computes the average of these minimum distances, and returns it as the result.

3.3.5 Steps sliding window

The last proposed recognition method is STEPS SLIDING WINDOW, SSW for short (Fig. 9).

This method is similar to the first one, namely WALK. After step segmentation, for each gallery walk to match, and before starting matching, the system assumes the longer sequence between the probe and the gallery walk as the main stream and uses the other one



**Fig. 9** Graphic example of STEPS SLIDING WINDOW method: the shorter signal slides over the longer ones taking the step starting points as overlap reference

as a sliding window. The sliding window is aligned using the starting points of steps on the main stream. Therefore, segmentation is computed but steps are not separated. Matching is performed over the overlapping region of the two signals. The returned distance will be the minimum of all DTW matchings performed while sliding.

This method, differently from the other ones that use steps segmentation, does not discard the outliers, since, as described above, segmentation results are only used for alignment purposes. We argue that a phenomenon similar to co-articulation in speech can hold for gait signals too, so that step segmentation might cause to lack some characteristic feature related to passing from one step to the following one. STEP SLIDING WINDOW does not suffer for the problem of a constrained number of steps, which is the limit of WALK, while retaining full information about step co-articulation, differently by the methods entailing step segmentation.

### 3.3.6 Considerations about algorithmic complexity

Concerning the algorithmic complexity, the following discussion regards the cost of a single match for each kind of algorithm. As for WALK, which is pure DTW, is  $O(n^2)$ , where  $n$  is the number of signal points (in the signals of our dataset, ten steps sum up to about 1250 points). Matching algorithms entailing the elimination of outliers as well as the identification of the centroid step from the probe template, require a main cycle that is  $O(m^2 p^2)$  to compare the steps in order to find out the outliers/centroid, where  $m$  is the average number of signal points in a single step (about 100 points in our non-interpolated signals) and  $p$  is the average number of steps (about 10 in our dataset). Further required computation has a lower complexity. However,  $n$  is about  $m * p$ , therefore, up to a multiplicative constant, this preliminary step has a complexity comparable to WALK. Then the comparison for each single step is  $O(m^2)$ , repeated for the number of steps to compare: only one comparison for BEST STEP,  $p^2$  comparisons on the average for ALL STEPS, taking again to  $O(n^2)$ . STEP SLIDING WINDOW complexity follows a different pattern. Given two walks to compare (probe and gallery) of  $n_1$  and  $n_2$  points respectively (corresponding to  $p_1$  and  $p_2$  steps), and assuming  $n_2$  as the shortest signal, we have an overall complexity of  $(p_1 - p_2 + 1) * O(n_2^2)$ . The larger the difference in signal length, the higher the number of comparisons, yet between shorter fragments. Given a signal of length  $n$ , the opposite situations are a signal to compare with a single step of length  $m$ , taking to  $p * O(m^2)$  comparisons, and two signal that are equal or differ for a single step, both taking to  $O(n^2)$  up to a multiplicative constant. Notice that the role of the two signals in the comparison, i.e., main stream and sliding window, depends on their relative length. Again, being  $n$  the length of the longest signal in the pair to compare,  $O(n^2)$  up to a multiplicative constant is an upper bound for the matching of a single pair of signals. In conclusion, the algorithmic complexity is quite the same for all methods. However, what changes from WALK to STEP SLIDING WINDOW is a progressive release of constraints, yet accompanied by a decrease in performance.

### 3.3.7 Algorithm to improve performances

As explained in Section 3.3, the system produces three distance values for each matching, one for each axis. A relevant point is to find the best way to combine these three results, since different accelerometers may acquire data with different relevance on the three axes. To do this, a dedicate algorithm is developed. This procedure tests the system trying all the different combinations of weights summing up to 1 on the three axes, with variations of 0.1. It should be carried out when significant variations happen in the exploited set of data

and is very useful because it can be used to reinforce accuracy on the dataset on which the system is currently working. Being a training procedure, is of course rather computationally demanding, but it is seldom carried out. When the dataset is very huge, the procedure can be executed on a smaller subset. In our experiments, we only carried it out for BWR dataset (about 30 minutes using two templates in the gallery and one as probe for each out of 30 users, using a notebook with a 2-nd generation I7 processor), but some experimental trials suggested to use the same values for the other datasets too.

## 4 Experiments setup

This section presents the tests performed with the recognition methods described in Section 3.3, in order to identify the better compromise between acquisition constraints and achieved accuracy. The proposed methodologies were tested on three different datasets, one acquired by us (called BWR and described in Section 3.0.1) and two publicly available ones, collected respectively by Zhang et al. (called from now on ZJU-gaitacc and described in Section 4.1.2) and Ngo and at. (called from now on OU-ISIR and described in Section 4.1.3). A further goal of the experiments carried out was the evaluation of the proposed step segmentation algorithm (in Section 3.2). The segmentation procedure described in [2] and the one in [15, 16], were reproduced and then tested using the proposed recognition methods, in order to evaluate the differences in the system performance. Detailed description of the compared algorithms can be found in Sections 4.2.2 and 4.2.1 respectively.

Section 4.1 describes in details all the datasets used in the experiments. Section 4.2 describes the segmentation procedures applied in the different test scenarios. Section 5 presents the experiment setup and the results obtained with all the dataset.

### 4.1 Datasets

This subsection presents a detailed description of the datasets used in testing experiments.

#### 4.1.1 BWR dataset

The BWR Dataset collects templates recorded with a OnePlus One smartphone (with an embedded LIS3DH Accelerometer,<sup>3</sup> with a sampling rate above 100 Hz). The acquisition procedure is the one described in Section 3.0.1. The dataset contains 3 or 4 walks of 30 subjects (22 males and 8 females) with a range of age between 16 and 65. In this dataset subjects wear different shoes but no one wears high heels. The users are asked to walk in a natural way for ten steps in a straight hallway. Templates are acquired detaching and repositioning the smartphone<sup>4</sup>, possibly on a different side of the belt. The Fig. 10 describes the BWR dataset archiving structure:

The DATA folder is created automatically by the system the first time a new dataset is loaded. It contains useful data extracted from each template (the *.csv* file) that can be computed once and for all.

#### 4.1.2 ZJU-gaitacc dataset

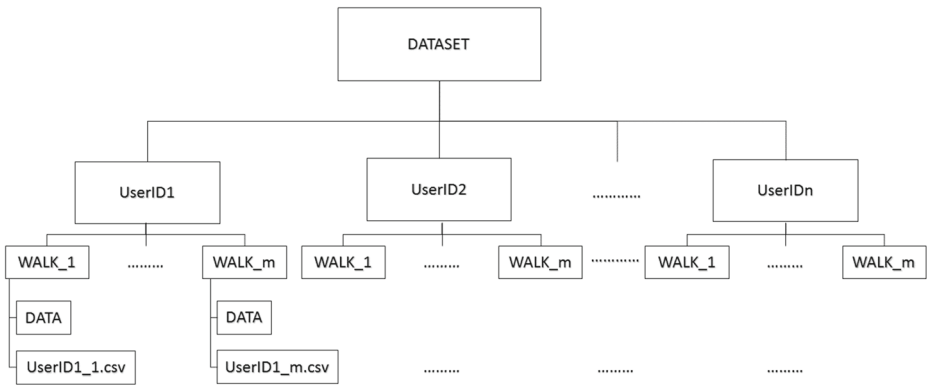
The ZJU-gaitacc dataset is related to the work described in [19]. It is composed by accelerometer signals from 5 Wii-Mote controllers (from the well-known Wii console)

---

<sup>3</sup><https://www.adafruit.com/datasheets/LIS3DH.pdf>

<sup>4</sup>Interested researchers can ask for this dataset by sending an email to authors.





**Fig. 10** Graphic representation of dataset structure

positioned at different body locations. The recordings are taken on level floor of 20m length. The tests carried out for our work take into account only data from the sensor positioned in the pelvis zone, that is more or less the same position used in BWR Dataset data acquisition. Moreover, these data are the ones on which the authors achieve the best performance in single accelerometer tests. The dataset contains data acquired from 175 users in 2 sessions, each one with 6 recordings. However, only 153 users out of 175 are used in our work, because the other 22 have only data from a single session (as described in the dataset information). It is worth underlining that gait data from this dataset has been previously interpolated and acquisition is done using an accelerometer with a good sampling rate (about 97 Hz) but a poor sensibility. This produces, in some cases, anomalous results, yet useful to highlight some specific behavior.

#### 4.1.3 OU-ISIR biometric database - inertial sensor dataset

The OU-ISIR inertial sensor dataset was acquired by Ngo Thanh Trung et al. and introduced in [11]. It collects data from 744 subjects, acquired using 3 IMUZ with KXTF9 accelerometers and gyroscope, and a Motorola ME860 smartphone equipped with an accelerometer. This gives to the researchers the access to data from 3 different positions (left and right hip and tailbone zone for IMUZs and tailbone zone for the smartphone). The recordings contain data from 12 seconds of walk, divided into four files, two with level floor, and one respectively for slope up and slope down (9m for the floor level and 3m for the slopes one). This means that the setting of the capture procedure is always the same, e.g., the device are never detached and reattached to the same user. For each subject, data belongs to one session only. The templates collect information from accelerometers and gyroscopes. Our tests only use the accelerometer data from level walk, taking left and right IMUZ signals as they were two different walks; therefore, the subset used for our experiments contains four segments for each subject, actually all taken from the same walk, two from each side IMUZ accelerometer; this is done in order to have a consistent comparison with the position of BWR dataset, as done for the ZJU-gaitacc dataset, and also to maintain a acceptable number of different signals per subject.

Two points are worth noticing. First of all, complete data from all the 744 subjects is available only for the tailbone zone IMUZ sensor (that they called IMUZ\_CENTER). Data acquired from all four sensors are only from 495 subjects. As a second point, with respect to the walk templates from BWR Dataset and ZJU-gaitacc Dataset, the ones provided by

OU-ISIR Dataset are significantly shorter (the number of signal points is about a third of the other two datasets). We argue that having such reduced number of signal points, while speeding up the matching, may, on the other hand, negatively affect recognition accuracy due to the higher weight gained by spike variation. However, a challenging aspect of the dataset is the variation in subject age (from 4 to 79 year old).

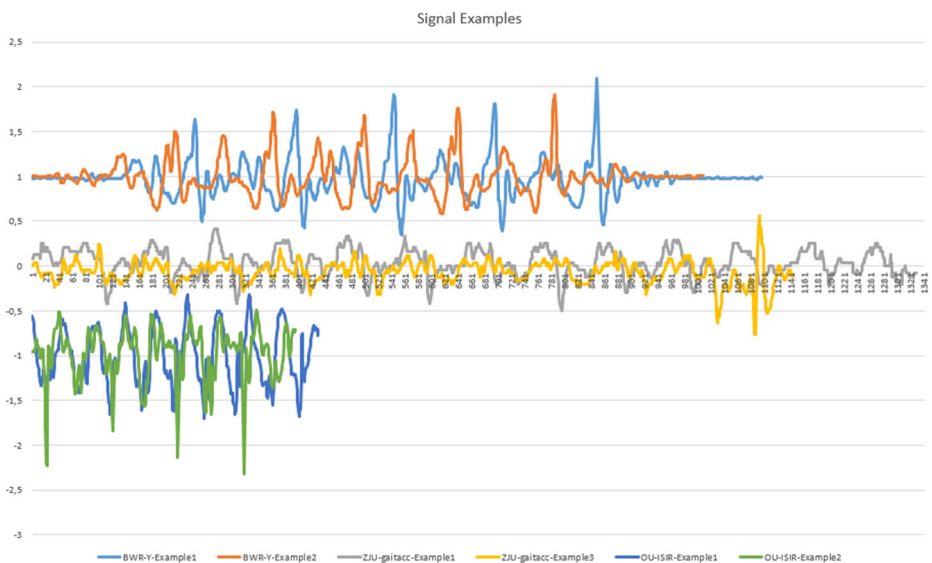
#### 4.1.4 Notes and discussion on the used datasets

The choice of the two public datasets to use together with BWR is not accidental. The first reason is their large amount of data on which it is possible to test the performances of the various algorithms, both for step segmentation and recognition. The second concerns their substantial differences: the ZJU-gaitacc provides a huge number of templates from the same user (12 walks per user recorded in 2 sessions) but less subjects (even if 175 is not a so low number) while the OU-ISIR collects data from a much higher number of subjects, but they came from only one session, the different walks are actually segments of the same walk, and their length is very short (the 9m floor walk at 100Hz of sample rate provides about 400 samples for this gait signals vs. more than 1200 samples in BWR and ZJU-gaitacc). Furthermore, the alignment of the collected time series present significant differences, and ZJU-gaitacc signals are also interpolated. Figure 11 shows two examples per dataset, that highlight these point. Therefore, achieving satisfying results on all of them is quite challenging.

In order to use these datasets, we exploited a dedicated procedure to convert the data archives into a common format, namely the one described in Section 3.0.1.

## 4.2 Compared step segmentation procedures

This subsection presents two cycle detection and segmentation algorithms proposed respectively by Rong et al. in [15, 16] (in the following, Rong algorithm for short) and Derawi et al. in [2] (in the following, Derawi algorithm for short).



**Fig. 11** Examples of signals from the three used datasets

### 4.2.1 The Rong algorithm

As for the step segmentation algorithm proposed in Section 3.2, the segmentation procedure proposed by Rong et al. is carried out on the most significant axis and then projected on the other two. In their work, the accelerometer in the acquisition of their data is attached on the hip. In all the three datasets used for our experiments, data are from the accelerometer on the hip, so that Y is to be assumed as the principal axis, and all segmentation algorithms are applied on it and results projected on the other two ones. The phases of their procedure are reported below:

1. normalization of the acceleration data to  $[-1,1]$ ;
2. noise elimination through wavelet denoising;
3. search for the local minimum points;
4. search for the first value with a different sign after the local minimum points; these points are the beginning or the end of a step;
5. segmentation: create a gait cycle with every four consecutive points in this set.

Since our algorithm do not use noise elimination, the comparison was stressed by skiping phase 2 of Rong method. In this way this method is by far the fastest one, though the results are lower than those achieved by the other two ones (see Section 5).

### 4.2.2 The Derawi algorithm

A brief explanation of the cycle detection and segmentation algorithm proposed by Derawi et al. is sketched below.

#### Detection:

1. choose a random sequence of 70 values in the middle of the walk signal;
2. compute a distance array, using DTW, containing distances between the chosen sequence and all sequences of 70 elements in the entire walk;
3. look for the minima in this distance array;
4. compute an average cycle length ( $\gamma$ ) using the distances between all consecutive found minima.

#### Segmentation:

1. start from a local minimum in the middle of the walk signal ( $P_{start}$ );
2. look for local minima near  $P_{start} + \gamma$  and  $P_{start} - \gamma$  using Neighbor Search;
3. repeat in both directions until end and start of signal respectively .

This algorithm is the slowest of the three tested ones, yet being the most challenging competitor for our proposal (see Section 5).

## 5 Results and discussion

For better readability, the experimental results are grouped in tables according to the benchmark datasets, and for each table a section is devoted to each recognition modality. Each section in Table 1 shows the results divided by matching method (rows) and segmentation algorithm (columns) for BWR dataset. Tables 2 and 3 are similarly organized to show the results for ZJU-gaitacc and OU-ISIR datasets respectively.

**Table 1** Results by matching method (rows) and segmentation algorithm (columns) for BWR dataset

Dataset: BWR-Verification Single Template (In terms of EER)					
	BWR	RONG 2007	DERAWI 2010	Differences	
Walk	0.1836			NA	
Best Step	0.3064	0.4891	0.3430	-0.1827	-0.0366
Best Step VS All	0.2825	0.3100	0.3202	-0.0275	-0.0377
All Steps VS All	0.2019	0.5117	0.3155	-0.3098	-0.1136
Step Sliding Window	0.2158	0.2754	0.1942	-0.0596	0.0216
Dataset: BWR-Verification Multiple Template (In terms of EER)					
Walk	0.1477			NA	
Best Step	0.3356	0.37495	0.33	-0.0394	0.0056
Best Step VS All	0.297	0.34685	0.369	-0.0499	-0.0720
All Steps VS All	0.19	0.35435	0.32	-0.1644	-0.1300
Step Sliding Window	0.22	0.3373	0.2	-0.1173	0.0200
Dataset: BWR-Open Set (In terms of EER)					
Walk	0.32445			NA	
Best Step	0.6383	0.734	0.623	-0.0957	0.0153
Best Step VS All	0.6702	0.7553	0.7021	-0.0851	-0.0319
All Steps VS All	0.4468	0.5426	0.6064	-0.0958	-0.1596
Step Sliding Window	0.5426	0.7128	0.4149	-0.1702	0.1277
Dataset: BWR-Recognition Rate					
Walk	0.8936			NA	
Best Step	0.4149	0.3511	0.4042	0.0638	0.0107
Best Step VS All	0.4362	0.4255	0.3404	0.0107	0.0958
All Steps VS All	0.6489	0.5638	0.4574	0.0851	0.1915
Step Sliding Window	0.5851	0.385	0.7553	0.2021	-0.1702

Each table section refers to a different recognition setting

In this work, we used different protocols to evaluate the performance of the system than the one used in [1]. Details of the new protocols are described next.

Recognition was tested in four different settings: verification with a single template per subject in the gallery (performance is measured by an all-against-all matching), verification with multiple templates per subject in the gallery (the best matching <probe-subject gallery template> is returned), open set identification with multiple templates per subject in the gallery (each probe is considered to belong or not to the gallery in turn), and closed set identification with multiple templates per subject, using all-against-all matching (each walk is used as probe and matched against all the others walks in the dataset)

**Table 2** Results by matching method (rows) and segmentation algorithm (columns) for ZJU-gaitacc dataset

Dataset: ZJU-gaitacc - Verification Single Template (In terms of EER)

	BWR	RONG 2007	DERAWI 2010	Differences	
Walk	0.3269			NA	
Best Step	0.3402	0.481	0.4783	-0.1408	-0.1381
Best Step VS All	0.3702	0.479	0.4778	-0.1088	-0.1076
All Steps VS All	0.3476	0.481	0.4773	-0.1334	-0.1297
Step Sliding Window	0.3383	0.482	0.4812	-0.1437	-0.1429

Dataset: ZJU-gaitacc - Verification Multiple Template (In terms of EER)

Walk	0.0926			NA	
Best Step	0.328	0.302	0.2847	0.0260	0.0433
Best Step VS All	0.4104	0.335	0.3311	0.0754	0.0793
All Steps VS All	0.3625	0.275	0.2704	0.0875	0.0921
Step Sliding Window	0.1025	0.302	0.2923	-0.1995	-0.1898

Dataset: ZJU-gaitacc - Open Set (In terms of EER)

Walk	0.3233			NA	
Best Step	0.4682	0.568	0.484	-0.0998	-0.0158
Best Step VS All	0.5726	0.644	0.5394	-0.0714	0.0332
All Steps VS All	0.5397	0.467	0.4231	0.0727	0.1166
Step Sliding Window	0.4162	0.579	0.5608	-0.1628	-0.1446

Dataset: ZJU-gaitacc - Recognition Rate

Walk	0.9282			NA	
Best Step	0.8274	0.5673	0.6824	0.2601	0.1450
Best Step VS All	0.6668	0.5058	0.6487	0.1610	0.0181
All Steps VS All	0.714	0.7196	0.8102	-0.0056	-0.0962
Step Sliding Window	0.7671	0.5366	0.5563	0.2305	0.2108

Each table section refers to a different recognition setting

except itself). As said, in the first verification setting, a pure all-against-all matching is carried out. The gallery is made up by only one template per user, so that the number of *GENUINE* tests is equal to  $\#templates$  and the number of *IMPOSTOR* tests is equal to  $\#templates * (\#templates - \#templates\_right\_user)$ , where  $\#templates\_right\_user$  is the number of templates that belong to the same user as the current probe. In case of the same number of templates per user (call it  $\#T$ ), the formula can be simplified into  $\#templates * (\#templates - \#T)$ . In the second verification setting, a gallery with multiple templates per user is considered, and an all-against-groups approach is exploited. During verification the best matching (lowest distance from the probe) for each group of templates belonging to the same user is returned. Of course matching of each template against itself is not considered. The number of *GENUINE* tests will be equal to  $\#templates$  and the

**Table 3** Results by matching method (rows) and segmentation algorithm (columns) for OUI-ISIR dataset

Dataset: OU-ISIR - Verification Single Template (In terms of EER)					
	BWR	RONG 2007	DERAWI 2010	Differences	
Walk	0.3661			NA	
Best Step	0.4405	0.4243	0.4115	0.0162	0.0290
Best Step VS All	0.4535	0.4333	0.4288	0.0202	0.0247
All Steps VS All	0.4472	0.4044	0.414	0.0428	0.0332
Step Sliding Window	0.3675	0.3678	0.3625	-0.0003	0.0050
Dataset: OU-ISIR - Verification Multiple Template (In terms of EER)					
Walk	0.2723			NA	
Best Step	0.4116	0.379	0.3575	0.0326	0.0541
Best Step VS All	0.3942	0.3553	0.3382	0.0389	0.0560
All Steps VS All	0.396	0.3054	0.3356	0.0906	0.0604
Step Sliding Window	0.2722	0.2734	0.2714	-0.0012	0.0008
Dataset: OU-ISIR - Open Set (In terms of EER)					
Walk	0.7962			NA	
Best Step	0.821	0.9022	0.7942	-0.0812	0.0268
Best Step VS All	0.8372	0.9022	0.7942	-0.1065	0.0683
All Steps VS All	0.798	0.7958	0.7496	0.0022	0.0484
Step Sliding Window	0.8003	0.8008	0.7976	-0.0005	0.0027
Dataset: OU-ISIR - Recognition Rate					
Walk	0.2381			NA	
Best Step	0.2422	0.1111	0.2173	0.1311	0.0249
Best Step VS All	0.2386	0.1408	0.2673	0.0978	-0.0287
All Steps VS All	0.2750	0.2407	0.2715	0.0343	0.0035
Step Sliding Window	0.2355	0.2320	0.2376	0.0036	-0.0021

Each table section refers to a different recognition setting

number of *IMPOSTOR* tests will be equals to  $\#templates * (\#users - 1)$ . In open set identification setting, each probe is considered to belong or not to a subject in the gallery in turn, therefore both the number of *GENUINE* tests and the number of *IMPOSTOR* tests will be equal to  $\#templates$ . In closed set identification, no *GENUINE/IMPOSTOR* distinction is made, and there is simply a number of tests equal to  $\#templates$ .

As expected, the tables show a general trend towards an increase in performances when using more gallery templates per user. This especially holds for ZJU-gaitacc dataset, where each subject has up to 11 templates in the gallery, since this allows a higher robustness to intra-personal variations. As a matter of fact, this consideration is widely accepted in literature [7].

As for the first three recognition settings, results are in terms of EER, while for the last one we report RR and later we show some further results using CMC curves. For sake of space, it is not possible to report all results in terms of performance curves. Tables further report the gain/loss of our segmentation algorithm with respect to the compared ones (column Differences with differences w.r.t. Rong and Derawi algorithms respectively). In order to provide a first-sight impression, gains are in green (lighter) cells and losses in red (darker) ones. It is worth reminding that Rong algorithm is reproduced without the denoising phase, which is not present in either BWR or Derawi algorithms, and is therefore deliberately avoided.

## 5.1 Results for BWR dataset

The different sections of the Table 1 refer to the different recognition settings for BWR dataset. For each section, the results are indexed by matching method (rows) and segmentation algorithm (columns).

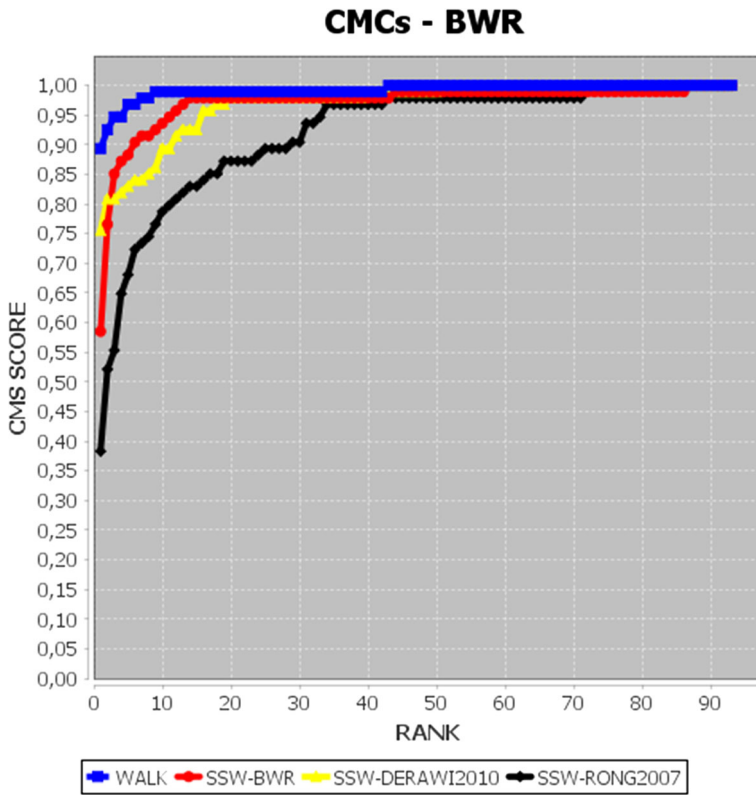
Starting from verification results, the respective table sections show a less significant improvement w.r.t. the other datasets (see below) when passing from a single template to multiple templates in the gallery, but this depends on the fact that BWR dataset only includes two/three templates per user in the gallery. For this reason, the advantage is less evident. On the other hand, it is possible to observe a general improvement achieved by BWR step segmentation w.r.t. both Rong and Derawi algorithms. In particular, the gain is systematic compared to Rong, while in some cases (three out of eight) there is a slight loss compared to Derawi. This is a very promising result, since it suggests that if some denoising as well as interpolation operations were carried out in advance on the signal, the results would be even better. In the present implementation however, the best results are achieved without any segmentation, i.e., by the WALK method. For this matching method EER is 0.18 in single-template setting and  $\approx 0.15$  in multiple-template setting. As it will be shown in the following, this is a general trend also for the other recognition settings and for the other datasets, and testifies that step segmentation is a critical operation that can affect the following results. As mentioned above, it is possible to argue that this can be due to a kind of co-articulation effect between successive steps that is not accounted for when segmenting steps. As for single-template verification for BWR dataset, the best result using segmentation is achieved by the Derawi algorithm with STEP SLIDING WINDOW matching (EER is 0.19) followed by BWR segmentation with ALL VS. ALL matching (EER is 0.20). However, a general trend sees a ranking with ALL VS. ALL being better than STEP SLIDING WINDOW. It would have been reasonable to expect the inverse, due to the higher similarity of STEP SLIDING WINDOW with the best WALK. As a matter of fact, this happens when using Rong and especially Derawi. These two results suggests that the STEP SLIDING WINDOW matching method is very promising but must be further investigated.

As expected, when passing to open set identification there is a general fall of performance. The matching distance from the right subject must be below the acceptance threshold, but also be the smallest one. EER achieved by WALK is about doubled w.r.t. to verification (EER is 0.32), and doubled again when introducing step segmentation. The comparison between BWR segmentation and Rong and Derawi confirms the superiority over Rong, while the comparison with Derawi produces a balanced result. Even in this case, the best result with BWR segmentation is achieved by ALL VS. ALL matching (EER is  $\approx 0.45$ ), followed by STEP SLIDING WINDOW (EER is  $\approx 0.54$ ). However, when using Derawi segmentation, it is possible to achieve the best result after WALK, provided by STEP SLIDING WINDOW (EER is 0.41).

Closed set identification confirms the general trend for this dataset observed for the other recognition settings. BWR segmentation overcomes Rong in all cases, and Derawi in all cases except one (when using STEP SLIDING WINDOW). WALK achieves the best RR (0.89), followed by ALL VS. ALL matching, and by STEP SLIDING WINDOW, except in the case of Derawi segmentation with which STEP SLIDING WINDOW achieves the second best result after WALK (0.75). Finally, Fig. 12 shows the CMC curves obtained for WALK, for the second best matching in terms of RR, namely STEP SLIDING WINDOW when using Derawi segmentation, and for the same matching method using the other step segmentations. It is interesting to notice that BWR segmentation curve, though showing a lower RR, presents a better slope and this suggest once more that the method can be significantly improved.

### 5.2 Result for ZJU-gaitacc dataset

Table 2 shows the results of the tests for ZJU-gaitacc dataset. Each section of the table refers to a different recognition setting, and the results are indexed by matching method (rows) and segmentation algorithm (columns).



**Fig. 12** CMC curves achieved on BWR dataset by WALK, by the second best matching, STEP SLIDING WINDOW when using Derawi segmentation, and for the same matching method using the other step segmentations



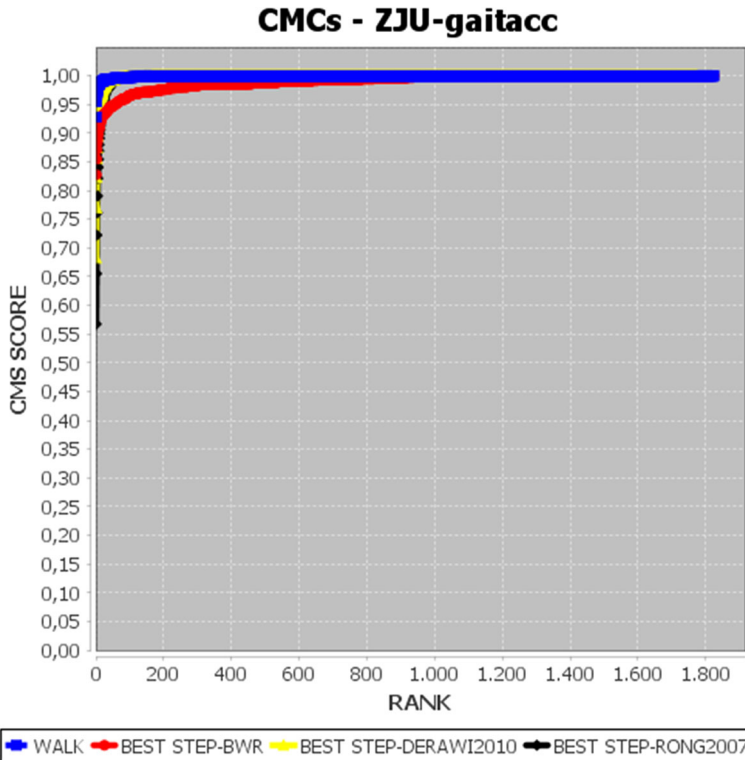
Starting again from verification settings, it is possible to notice that for this dataset the difference between single-template and multiple-template identification is much more significant, but especially when using the unsegmented signals, i.e. WALK matching (EER is  $\approx 0.33$  for single-template verification and  $\approx 0.09$  for multiple-template verification). When segmentation is entailed, the differences present a smoother trend. On one side, this demonstrates the advantage of having more templates per subject in the system gallery. On the other side, it further confirms the critical role of step segmentation. In single-template verification, BWR segmentation overcomes the other two methods, with the best results achieved by STEP SLIDING WINDOW (EER is  $\approx 0.34$ ). However, in multiple-template setting, it always achieves slightly lower performance, except when STEP SLIDING WINDOW is exploited. In this case, it provides the second best result after WALK matching (EER is 0.1). Except for this case, the ranking among matching methods entailing segmentation is less sharp than with BWR dataset. ALL VS. ALL matching is still generally the best using segmentation, but for this dataset the second place is sometimes occupied by BEST STEP, that requires to only match the centroid steps of two walk signals.

It is interesting to notice that this time, when passing to open set identification, the results fall again but in a less dramatic way (EER is 0.32), except for the significant difference with the result achieved by the unsegmented signal in multiple-template verification. This may be ascribed to the beneficial effect of having a larger number of templates per subject that better supports open set identification too. Furthermore, results are better on the average than those achieved with the much smaller BWR dataset. As for segmentation algorithms, the comparisons shows an alternating trend, but is its interesting to notice that the second best result is achieved again by STEP SLIDING WINDOW with BWR segmentation (EER is 0.41). This is a very encouraging result.

Closed set identification achieves a RR by WALK matching ( $\approx 0.93$ ) that is surprisingly higher than achieved for the much smaller BWR dataset (0.89). The same counterintuitive trend can be observed on average for all results in the table. BWR segmentation provides better matching results than the other two except when using ALL VS. ALL matching. The best result when using segmentation is achieved by BEST STEP after BWR segmentation, entailing centroid matching (RR is  $\approx 0.83$ ), followed by ALL STEPS VS. ALL after Derawi segmentation (RR is 0.81).

The fact of achieving results that are often better than those achieved for BWR dataset, seems to also confirm that interpolation, which is carried out on ZJU-gaitacc, can provide beneficial effects on final recognition. Furthermore, the repeated good results achieved using BEST STEP on this dataset suggest to investigate a smarter way to exploit centroids steps extracted from the the walk signal. Figure 13 shows the CMC curves obtained for WALK, for the second best matching in terms of RR, namely BEST STEP when using BWR segmentation, and for the same matching method using the other step segmentations. It is interesting to notice that Derawi segmentation curve, though showing a lower RR, presents a better slope.

In the work [19], that introduces the ZJU-gaitacc dataset, the results obtained from pelvis accelerometer only are the ones that can be compared to our ones. The authors report a RR of 73.4 % and a EER of 8.9 %, using their own method on the dataset signals (see Section 2). Our results on the same dataset appear to be slightly better, but the different experimental setting must be taken into account. As a matter of fact, for both identification and verification modalities, the templates from the two sessions are used either as probe or as gallery.



**Fig. 13** CMC curves achieved on ZJU-gaitacc dataset by WALK, by the second best matching, BEST STEP when using BWR segmentation, and for the same matching method using the other step segmentations

### 5.3 Result for OU-ISIR dataset

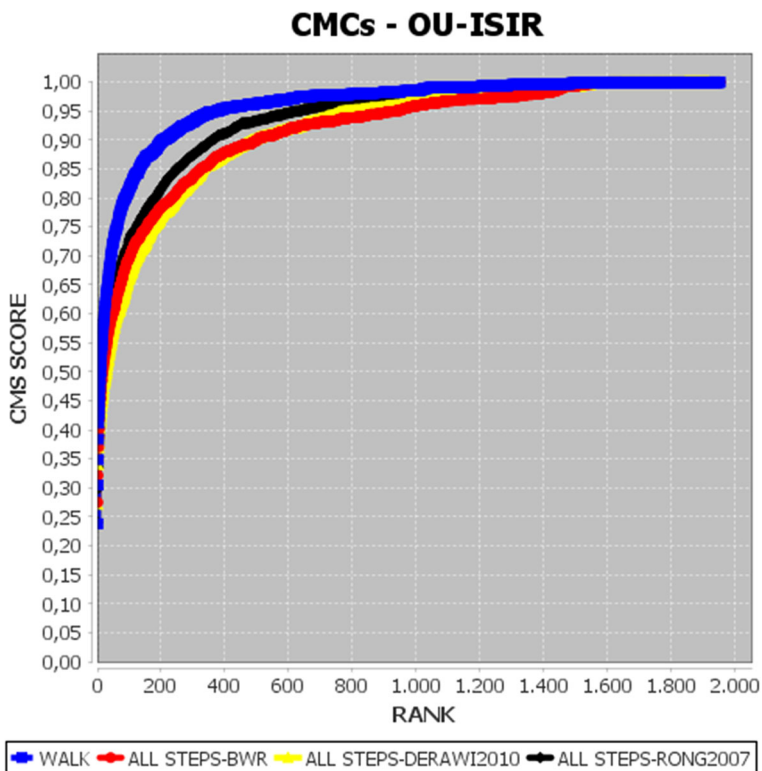
Table 3 shows the results for OU-ISIR dataset. The table is divided into sections that refer to the different recognition settings. Each section is indexed by matching method (rows) and segmentation algorithm (columns). It is worth reminding that, though collecting data from 744 subjects, those acquired from all sensors are only from 495 subjects. Therefore, since we carry out experiments on side sensors, this is the reduced set of subjects that we consider. Moreover, walks provided by OU-ISIR Dataset are significantly shorter, since the number of signal points is about a third of the walks in both BWR and ZJU-gaitacc, that on the contrary are quite similar in length. This is due to the fact that each original walk is divided into four segments, each treated as a different walk. Of these, we only considered level walks from both sides. It is also to point out that for 11 subjects it was not possible to use the side accelerometer data since they were null.

It is immediate to notice the significantly worse results achieved for this dataset. In particular, an element that really stands out is the much smoother difference between WALK and the matching methods entailing step segmentation. A reasonable explanation of this is the very short length of walks, so that segmentation actually has no real effect, while matching of the entire signal is more sensible to noise, and also sharp sudden yet temporary variations have a disruptive effect on matching. It is interesting to point out that closed set identification achieves extremely low results; verification performance too is lower than the

one achieved with the other datasets, even if not equally dramatically (even in this case the advantage of having more templates per subject still holds); finally open set identification achieves results that are well below a random decision. It is also worth underlining that the segmentation algorithm has little influence on the final recognition results. Though Derawi algorithm seems outstanding, looking at the actual differences it is possible to notice that they generally take lower values that with the other datasets.

Figure 14 shows the CMC curves obtained for WALK, for the best matching (for this dataset) in terms of RR, namely ALL STEPS VS. ALL when using BWR segmentation, and for the same matching method using the other step segmentations. It is interesting to notice that for this dataset WALK does not achieve the best RR, but its CMC curve has the best slope anyway. The same matching with BWR, though achieving the highest RR, has a CMC curve lower than that obtained with the algorithm that gave the worst results in general for the other datasets, i.e., Rong, while BWR CMC is comparable to that obtained using Derawi algorithm.

The comparisons reported in [11] entail four state-of-the-art methods in literature, among which Rong and Derawi ones, and no method is proposed by the authors themselves. First of all it is worth mentioning that the results are reported only for verification modality, in terms of EER and ROC curves. The results obtained by the complete application of Rong

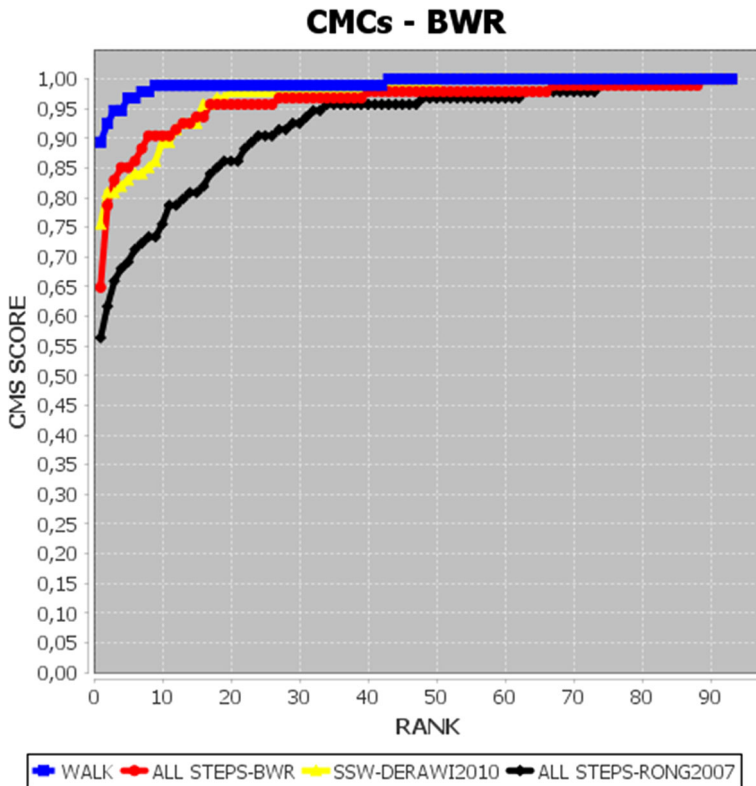


**Fig. 14** CMC curves achieved on OU-ISIR dataset by WALK, by the second best matching, BEST STEP when using BWR segmentation, and for the same matching method using the other step segmentations

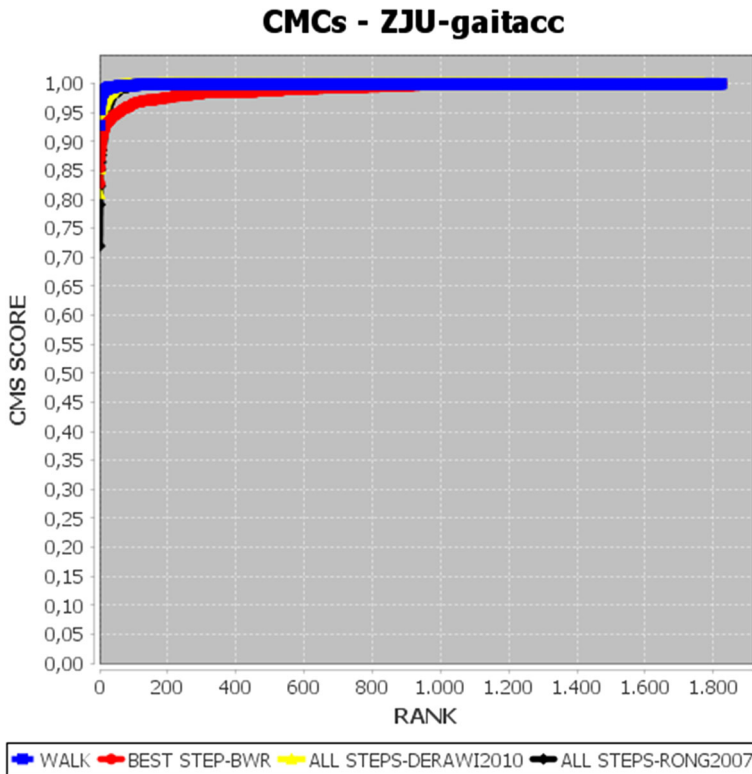
and Derawi methods (including both step segmentation and recognition) on the respective original datasets, in terms of EER, are respectively of 5.6 % and 5.7 %. EER with OU-ISIR dataset falls to 14 % for both methods. However, the reported comparison regards data from OU-ISIR taken from the accelerometer at the back waist. The position was the same for the experiments by Rong et al., but those by Derawi et al. exploited the accelerometer on the left hip. Ngo et al. themselves further assess the impact on performance of the accelerometer position, and the center IMUZ, namely the one at the back waist, is exactly the one with the worst ROC curve. The second worst is the one achieved by the smartphone, but in their setting even this device is positioned at the back waist.

In conclusion this dataset is surely very challenging, especially for the high number of subjects of different ages. However some issues arise when examining the dataset in depth, especially related to the low number of walks for each subject and for their very short length, or for the very low number of points in the signals at least.

It is clear from the above notes that without a common benchmark any kind of comparison is quite unreliable, therefore even taking further considerations from the results reported in Section 2 is quite useless.



**Fig. 15** Best CMC curves achieved on BWR dataset by each segmentation algorithm, with the indication of the matching strategy producing that result



**Fig. 16** Best CMC curves achieved on ZJU-gaitacc dataset by each segmentation algorithm, with the indication of the matching strategy producing that result

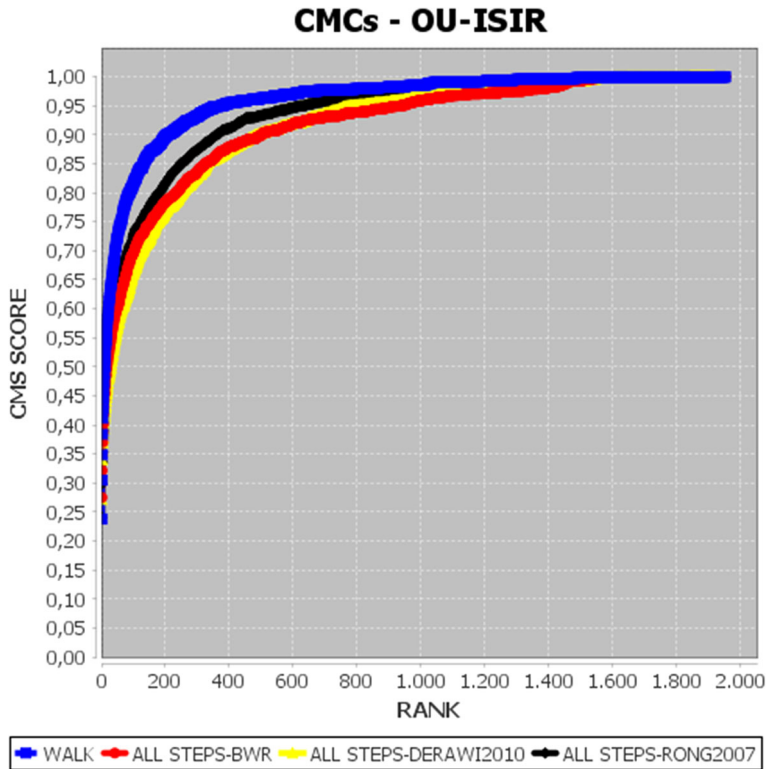
#### 5.4 Some further considerations about the compared segmentation algorithms

In order to have a better idea of the results of the comparison among the different step segmentation algorithms, the following Figs. 15, 16, and 17 show, for each dataset, the best CMC achieved by each segmentation algorithm with the indication of the matching used, in comparison with the CMC achieved by WALK.

Figure 15 shows that BWR segmentation on BWR dataset achieves better results than Derawi both for RR and CMC slope, while Rong algorithm shows a generally lower performance.

Figure 16 shows curves for ZJU-gaitacc. BWR segmentation achieves better results in terms of RR by using BEST STEP, but has the worst CMC slope, although the difference is not dramatically significant. The other two segmentation algorithms achieve their best performance with ALL STEPS VS ALL. This suggests to better investigate the use of centroid steps, and that methods comparing single steps achieve better accuracy on interpolated data.

Figure 17 shows the best results achieved by each segmentation algorithm on OU-ISIR dataset. The difference in performance is well evident, as well as the substantial equivalence of the segmentation algorithms. Also in this case, for all segmentation algorithms the



**Fig. 17** Best CMC curves achieved on OU-ISIR dataset by each segmentation algorithm, with the indication of the matching strategy producing that result

same matching method achieves the best performance, namely ALL STEPS VS ALL. As a consequence, in this case the figure is equal to Fig. 14.

The method that was supposed to achieve the best compromise between accuracy and acquisition constraints, namely STEP SLIDING WINDOW, does not appear in the above figures, meaning that it did not achieve remarkable performance. This indicates that further investigation is requested to improve it.

## 6 Conclusions and future work

This paper presented the schema of a complete system for gait recognition and showed the work done so far to implement the core recognition procedures. A novel segmentation algorithms and different matching strategies were tested, attempting to limit the number of constraints to data capture and support recognition "in the wild" in uncontrolled conditions. However, our assumption is that enrolling can still entail some control on the gait features. The achieved results are quite promising, even if, at present, the system exploits a very basic version of DTW. Good results were also achieved on a huge public dataset, which was captured in completely different conditions and pre-processed by interpolating signal data

(our dataset collects only raw data). The experimental results suggest some further closing considerations and guidelines for future work.

Unexpected good results were achieved on ZJU-gaitacc dataset compared with our much smaller BWR dataset. This especially holds when using 11 templates per subject in the gallery in the multiple template verification experiment. This demonstrates that recognition against a gallery collecting multiple templates per subject achieves better performance w.r.t. approaches using a single template. Having more gait signals, acquired in different sessions and in different conditions, allows to better address possible variations.

Another possible reason for the better results with ZJU-gaitacc dataset is that its data are interpolated, while BWR collects raw data instead. We plan to continue storing raw data in both BWR dataset and in a larger one that is being collected, in order to address different processing strategies. However, due to the mentioned result, it will be worth investigating the feasibility and real benefit of including an interpolation phase in the proposed walk processing algorithms.

As a further observation, it is to consider that in both BWR and OU-ISIR datasets, signals from right and left hip are mingled, while ZJU-gaitacc dataset, or better the subset that we used, seem to contain data from a fixed position on the same pelvis side. It is to consider that this may introduce more perturbations in the tests for the former two datasets, since, like the overall static bodily structure, also the dynamics of the two body sides might be slightly different.

The proposed segmentation algorithm processes raw data without any previous noise reduction. The better results achieved w.r.t. to the segmentation algorithm proposed by Rong et al. skipping this pre-processing operation has a twofold interpretation. From one side, they suggest that our segmentation algorithm can achieve even better results if some denoising is adopted. From the other side, they testify that it is possible to maintain the overall procedure less computationally demanding yet achieving satisfying results. The algorithm by Derawi et al. sometimes outperforms BWR segmentation, but has an higher computational time. It is worth underlining that, anyway, the best results are always obtained by matching unsegmented signals. As hypothesized, this can be ascribed to a kind of co-articulation influence between successive steps, which is a discriminant element that might be lost when using walk segments.

Few works in literature present experimental results for open set identification. Actually, this seems the most realistic identification scenario, where a limited white (or black) list just contains the set of subjects to recognize, while all the others must be rejected. However, it is reasonable to expect that this kind of recognition is the most challenging one, since the right template must be close enough to the probe, as in verification, but also the closest one to provide a correct identification. As a matter of fact, whatever the approach, using either segmented or unsegmented signals, and whatever the matching algorithm, the accuracy suffers for a significant decrease. In any case, it is to consider that gait is often classified as a soft biometrics. In the future, the proposed improvement will be to use our system also in a multibiometric setting. Its results will be fused with those from computer-vision based gait recognition, as well as other biometric modalities, to enforce identification for accessing critical locations/services. Visual recognition of soft features could also be used to limit the search space, by eliminating part of the gallery candidates by extracting bodily features from the probe, such as height or skin color.

The worse results were achieved by OU-ISIR dataset notwithstanding the segmentation algorithm or the matching strategy. A number of hypotheses can explain this. Of course, the most evident factor is the higher number of subjects as well as the higher variability in the age range. Even if the number is not that high for all sensors, it is higher than in

ZJU-gaitacc anyway. However, the number of templates per subject is lower than ZJU-gaitacc, and they are also much shorter than those in both the other datasets. This might increase the sensitiveness to noise.

Future developments of the proposed approach and related experiments will include gait signals acquired at different speeds on different way slopes, with different smartphones, and possibly different kinds of shoes, e.g., with high heels. At present, a dedicated app for the data acquisition is being developed. The next "steps" will entail finding a way to normalize gait data from different kinds of accelerometer models, that can present different offset and also different data ranges; a further future research will investigate an algorithm for step segmentation that does not need to know the number of steps even during enrollment. Last but not least, considering the results achieved with data from ZJU-gaitacc dataset, which are interpolated, the advantages of interpolation will be further investigated.

## References

1. De Marsico M, Mecca A (2015) Biometric walk recognizer. In: *New Trends in Image Analysis and Processing ICIAP 2015 Workshops*. Springer, pp 19–26
2. Derawi MO, Bours P, Holien K (2010) Improved cycle detection for accelerometer based gait authentication. In: *Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2010 Sixth International Conference on*, pp 312–317
3. Derawi MO, Nickel C, Bours P, Busch C (2010) Unobtrusive user-authentication on mobile phones using biometric gait recognition. In: *Intelligent Information Hiding and Multi-media Signal Processing (IIH-MSP), 2010 Sixth International Conference on*, pp 306–311
4. Gafurov D (2007) A survey of biometric gait recognition: Approaches, security and challenges. In: *Annual Norwegian Computer Science Conference*. Citeseer, pp 19–21
5. Gafurov D, Helkala K, Søndrol T (2006) Biometric gait authentication using accelerometer sensor. In: *Journal of computers* 1.7, pp 51–59
6. Gafurov D, Snekenes E, Bours P (2010) Improved gait recognition performance using cycle matching. In: *Advanced Information Networking and Applications Workshops (WAINA), 2010 IEEE 24th International Conference on*, pp 836–841
7. Jain AK, Ross A, Prabhakar S (2004) An introduction to biometric recognition. In: *Circuits and Systems for Video Technology*, IEEE Transactions on 14.1, pp 4–20
8. Juefei-Xu F, Bhagavatula C, Jaech A, Prasad U, Savvides M (2012) Gait-id on the move: Pace independent human identification using cell phone accelerometer dynamics. In: *Biometrics: Theory, Applications and Systems (BTAS), 2012 IEEE Fifth International Conference on*, pp 8–15
9. Lee TKM, Belkhatir M, Sanei S (2014) A comprehensive review of past and present vision-based techniques for gait recognition. In: *Multimedia tools and applications* 72.3, pp 2833–2869
10. Müller M (2007) *Information Retrieval for Music and Motion*. In: Springer Berlin Heidelberg, pp 69–84
11. Ngo TT, Makihara Y, Nagahara H, Mukaigawa Y, Yagi Y (2014) The largest inertial sensor-based gait database and performance evaluation of gait-based personal authentication. In: *Pattern Recognition* 47.1, pp 228–237
12. Ngo TT, Makihara Y, Nagahara H, Mukaigawa Y, Yasushi Y (2014) Orientation-compensative signal registration for owner authentication using an accelerometer. In: *IEICE TRANSACTIONS on Information and Systems* 97.3, pp 541–553
13. Nickel C, Busch C, Rangarajan S, Möbius M (2011) Using Hidden Markov Models for accelerometer-based biometric gait recognition. In: *Signal Processing and its Applications (CSPA), 2011 IEEE 7th International Colloquium on*, pp 58–63
14. Pan G, Zhang Y, Wu Z (2009) Accelerometer-based gait recognition via voting by signature points. In: *Electronics letters* 45.22, pp 1116–1118
15. Rong L, Jianzhong Z, Ming L, Xiangfeng H (2007) A wearable acceleration sensor system for gait recognition. In: *Industrial Electronics and Applications, 2007. ICIEA 2007. 2nd IEEE Conference on*, pp 2654–2659
16. Rong L, Zhiguo D, Jianzhong Z, Ming L (2007) Identification of individual walking patterns using gait acceleration. In: *Bioinformatics and Biomedical Engineering, 2007. ICBBE 2007. The 1st International Conference on*, pp 543–546



17. Sprager S, Juric MB (2015) Inertial Sensor-Based Gait Recognition: A Review. In: *Sensors* 15.9, pp 22089–22127
18. Trung NT, Makihara Y, Nagahara H, Sagawa R, Mukaigawa Y, Yagi Y (2011) Phase registration in a gallery improving gait authentication. In: *Biometrics (IJCB), 2011 International Joint Conference on*, pp 1–7
19. Zhang Y, Pan G, Jia K, Lu M, Wang Y, Wu Z (2015) Accelerometer-based gait recognition by sparse representation of signature points with clusters. In: *Cybernetics, IEEE Transactions on* 45.9, pp 1864–1875
20. Zhong Y, Deng Y (2014) Sensor orientation invariant mobile gait biometrics. In: *Biometrics (IJCB), 2014 IEEE International Joint Conference on*, pp 1–8



**Maria De Marsico** is Associate Professor at Sapienza University of Rome, Department of Computer Science, where she has been Assistant Professor from 2001 to 2015. She got her Master degree in Computer science from University of Salerno in 1988. Her scientific interests focus on Human Computer Interaction and Image Processing. Regarding the first one, she is especially interested in problems related to accessibility for users with special needs, and advanced techniques for personalized distance learning. Regarding the second one, she works on biometric recognition, including face, iris, gait, and multimodal recognition. She is Associate Editor of *Pattern Recognition Letters*, and Area Editor of the *IEEE Biometrics Compendium*. She published about 100 scientific works in international journals, conferences, and book chapters. She has been member of many Technical program Committees and is referee for several top journals. She recently organized the 2015 edition of CHIItaly, the biannual conference of the Italian chapter of ACM SIGCHI.



**Alessio Mecca** got his Master Degree in Computer Science from Sapienza University of Rome in 2015. He is currently a Ph.D Student in the same Department. His scientific interest regard biometrics and in particular gait recognition.