An adaptive and on-line IMU-based locomotion activity classification method using a triplet Markov model

Haoyu Li[a,*], Stéphane Derrode[a], Wojciech Pieczynski[b]

[a] *École Centrale de Lyon, Univ. Lyon*
LIRIS, CNRS UMR 5205, France.
[b] *Telecom SudParis, Univ. Paris-Saclay*
SAMOVAR, CNRS UMR 5157, France.

Abstract

Detecting locomotion activities is critical for the analysis of human daily activities. In this paper, an adaptive on-line classification method is proposed to detect four lower limb locomotion activities –walking, running, stair ascent and descent– from the signals of a unique wearable sensor. The method is based on a non-parametric triplet Markov model, to detect gait phases and activities simultaneously, in an unsupervised way. This capability allows the model to work at run-time, and so to be used on-line. Also, an algorithm that adapts model parameters suits for a wide range of healthy human is presented. From this adjustment ability, an initial model can gradually approach to the dedicated activity patterns. Experimental results with ten healthy subjects show that our algorithm can reach an overall classification accuracy up to 99.20%, after the stabilization of parameters adjustment, regardless of the users' gender, height, activity speed... Overall, the proposed algorithm presents a good performance in on-line parameters learning and high accuracy in classifying lower limb locomotion activities from a fount-mounted inertial measurement unit-based wearable sensor.

Keywords: Locomotion activity; Wearable sensor; On-line activity recognition; Triplet Markov chain; Adaptive iterative conditional estimation.

*Corresponding author
Email addresses: haoyu.li@ec-lyon.fr (Haoyu Li), stephane.derrode@ec-lyon.fr (Stéphane Derrode), wojciech.pieczynski@telecom-sudparis.eu (Wojciech Pieczynski)

1. Introduction

Locomotion activity has generated a high level of interest in recent years because of its large number of applications in many research fields, *e.g.* surveillance systems, health-care for elderly or sick people [1, 2] , rehabilitation [3] , daily life suggestion, and management... Wearable exoskeleton robotics may assist injured people to do exercises or daily activities such as walking and stair climbing [4]. [5] presented that gait analysis can be used in detecting gait abnormalities of disabled people. Of all these applications, those based on lower limb activities are among the most numerous [6, 7], including walking, running, stair ascent and descent, cycling, making turn...

Many different types of sensors have already been used to analyze locomotion activities. Among them, inertial measurement units (IMUs) are small and can be used to obtain kinematic information, such as acceleration and angular rate, of the body part where the sensor is placed: lower limbs, waist, chest, or arm... San-Segundo *et al.* [8] proposed a human activity recognition and segmentation system based on hidden Markov models (HMMs) for recognizing and segmenting six activities: walking, sitting, standing, lying, stair ascent and descent. The authors in [9] used data collected by smartphone and showed that human kinematics convey important information about user identity from motion patterns. The survey [10] reported that other types of sensors like camera, WIFI, Bluetooth, or electromyogram sensor can also be used for locomotion activities. Unfortunately, camera and wireless sensors need pre-installed cameras and beacons, and this results in the inability to use them out of the area of the pre-installed instruments. Electromyogram sensors need to put many nodes with cables on the body, which is not convenient for daily life. While IMU has no such restrictions or inconvenience, allowing to be used almost anywhere, even outdoors.

Given the many advantages of IMU sensors and their convenience, our goal is to identify four kinds of activity, namely walking, running, stair ascent and descent, from only one single foot-mounted IMU sensor. These activities share the feature in common: they are built with the same gait cycle with four phases named stance, push-up, swing and step-down. The only difference relies on the patterns and speed varies of the generated signals. Therefore, the duration of each gait phase and the signal shape with respect to specific activity will be different.

Generally, in order to classify the considered activities, segmentation is firstly performed to detect each gait cycle. Then it is possible to evaluate the signals within one gait cycle to determine the corresponding activity [11]. With this strategy, the classification results are extremely dependent on the accuracy of the gait cycle analysis. In this work, we develop a specific "triplet Markov

chain" (TMC) model [12] suited to classify the four activities. This model provides us with a way to simultaneously detect gait cycles (whatever the activity) and the activities themselves, by introducing two hidden sequences into the model. In this work, our claimed contributions are:

- A specific TMC model is designed to mimic the natural gait cycles and activity switches by introducing a cyclic left-right state transition graph, which allows detecting gait cycles and activity modes simultaneously.
- A non-parametric TMC model, called TMC-HIST, is proposed to better represent the distribution of observations conditionally to activities and gait phases, which is based on the histograms of historical data.
- An adaptive and on-line algorithm, based on TMC-HIST, is presented to allow auto-calibration (i) to the subject, whatever his/her gender, size, weight..., (ii) but also to the change of activities' pattern and road conditions at run time.

The remaining of the paper is organized as follows. Section 2 presents the off-line classification algorithm, the properties of activities and gait phases are illustrated first, and then the details of TMC-HIST is described, including the parameters learning method. Section 3 shows how it is possible to make the algorithm adaptive to users and for its use in real-life conditions. Section 4 describes the experiments conducted with a group of ten subjects. Section 5 discusses the performance of the proposed algorithm compared to the state-of-the-art works, a deep discussion on the behavior of the algorithm is also proposed there. Finally, conclusions and future work are proposed in the last section.

2. The off-line recognition algorithm

We start by describing the property and shape of the IMU signals to be used for activity classification. Then, we present the novel cyclic left-right TMC model with non-parametric modelling of data-driven densities by normalized histograms. The section ends with the calculations –required in an off-line scenario, also called "batch mode"– to learn model's parameters and to make the locomotion activity recognition algorithm unsupervised.

2.1. Activities and Gait Phases

The IMU sensor[1] is fixed on the shoe of the right foot as shown in Fig. 1, with X-axis of the sensor pointing to right, Y-axis pointing ahead and Z-axis up. For not mixing with the symbols of stochastic process in the remaining of the paper, the three axes will be represented as $\mathcal{X}$-axis, $\mathcal{Y}$-axis, $\mathcal{Z}$-axis, respectively. From the 9-DOF (degree of freedom) of the sensor (acceleration, angular rate and magnetic field, all in 3D), we will only use angular rates $[\omega^{\mathcal{X}}, \omega^{\mathcal{Y}}, \omega^{\mathcal{Z}}]$. The reason not considering acceleration is that the shoe and foot shape vary from person to person, especially for the front slope where the sensor is placed, thus the acceleration readings may be significantly different in each sensor axis. There are two reasons for not considering magnetic field. The first one is that it is used for estimating the orientation of the sensor. However in this work, we only focus on what kind of activity a person is performing, not where the person is heading for. Another reason is that magnetic field is very easily distorted by other unpredictable magnetic sources, such as buildings, electricity devices, cars...

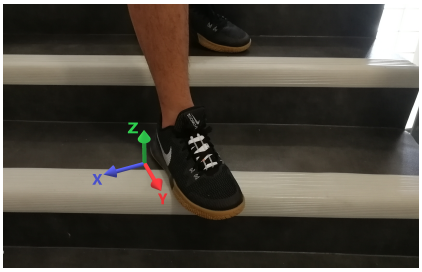


Figure 1: The placement of the IMU sensor on a shoe.

A quite large amount of methods have been used in investigating locomotion activities from IMU sensors, and we are going to only focus on works reported recently. Wu *et al.* [13] investigated the performance of imbalanced dataset for human activity recognition, their method showed better results compared to the existing methods. Traditional classifiers like Support Vector Machine (SVM) [14], Gaussian Mixture Model (GMM) [15] are commonly applied in locomotion activity detection. Authors in [16] also used several features and classifiers to test their impacts on recognition accuracy, with the help of acceleration of 9 sensors on the body. They proved that detecting par-

[1]Shimmer3 GSR+, more details at the manufacturer's site `http://www.shimmersensing.com/images/uploads/docs/ConsensysPRO_Spec_Sheet_v1.1.0.pdf`

ticular locomotion activities accurately needs specific features and classifiers. Their results showed that mean and standard deviation features provided the best accuracy out of all features evaluated by both KNN (K-nearest neighbor) and ensemble methods, while spectral entropy produced the worst performance. They also concluded that data pre-processing has nearly no impact on classifying accuracy. Also of interest, Wen *et al.* [17] proposed an AdaBoost-based algorithm to adapt and refine the model at run-time, by automatically selecting the most discriminating features. Their results were tested on several smartphone data-sets and showed significant improvement in recognition performance. Except for traditional classifiers, machine learning or deep learning techniques have also been investigated to recognize activities. Hassan *et al.* [6] proposed a method for activities training and recognition by using Deep Belief Network (DBN) and obtained an overall accuracy of 95.85% among 12 activities on smartphone sensor data. Whereas Ignatov [18] used Convolutional Neural Networks (CNN) to recognize 6 human activities, including 3 lower limb activities, in real-time from accelerometers. This method obtained an accuracy higher than 97%, and achieved an overall accuracy of 82.76% on a cross-dataset experiment.

The four activities we consider in this paper share a similar periodic pattern: from attaching to the ground to swinging in the air and then attaching to the ground again. This periodic pattern is called gait cycle and generally there are two common ways for segmenting each gait cycle:

(1) One simplest way is that one gait cycle is divided into two gait phases [19] , namely stance phase and swing phase.

(2) Based on the two gait phases in the first approach, we can divide the gait cycle into more detailed phases. As introduced in [20] , we utilize four gait phases in one gait cycle, *i.e.* stance, push-up, swing and step down.

The researchers Wang *et al.* did a lot of researches on activity recognition and the related applications using inertial sensors. Among their researches, [21] proposed an equestrian sports analysis based on the joint angle of rider's lower limb and the gait cycle of the horse, [22] proposed a pedestrian trajectory reconstruction method by detecting the stance phase when walking. These algorithms obtained good results and showed the great possible applications of gait analysis. However, the first research was working on the entire gait cycle without employing the gait phases, and the second one only focused on detecting stance gait phase of walking, not on all the gait phases and multiple activities. Martinez-Hernandez *et al.* [11] developed an adaptive Bayesian inference

system using three sensors placed on leg to recognize walking on different road conditions, *i.e.* level-ground, ramp ascent and descent. They introduced gait phases and attempted to recognize activities and gait phases simultaneously. The high accuracy (99.87%) indicates that gait phases can significantly improve the accuracy for walking activity. Inspired by the previous work, our goal is to extend the gait cycle to more general cases. Like the scheme of walking gait cycle shown in Fig. 2, apparently similar behaviors hold for all the other three activities.

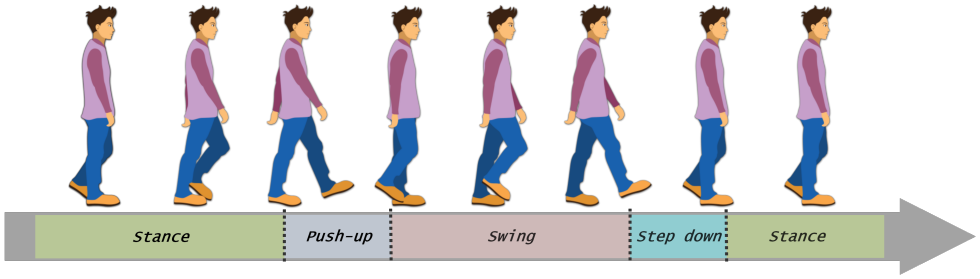


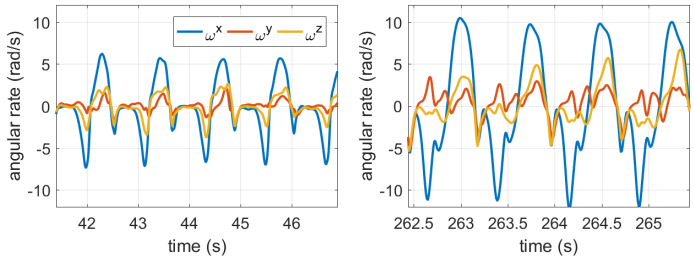
Figure 2: Right foot gait phases of walking cycle: push-up → swing → step down → stance. Similar gait cycle can be deduced for the three other activities, the only difference relying in the duration and signal shape of each gait phase.

The angular rates of the three sensor axes are quite different for different activities (*cf.* Fig. 3), hence providing a way to distinguish them correctly. In the off-line scenario, the entire data are acquired first, then, activities and gait phases are estimated using a parameter learning method, detailed in the next sections.

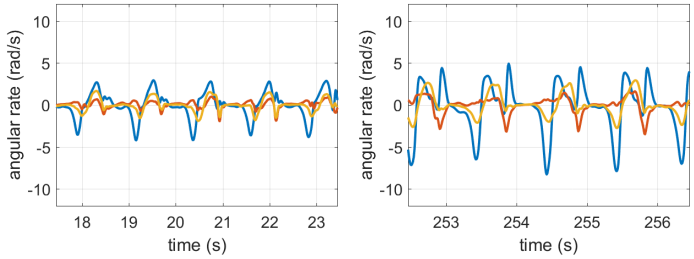
2.2. Non-parametric TMC

Consider a first discrete stochastic process $\boldsymbol{X} = (X_1, \ldots, X_N)$, each X_n, $n \in \{1, \ldots, N\}$, takes its values in $\Psi = \{1, 2, 3, 4\}$. The four values in Ψ respectively represent walking, running, stair ascent and descent. In a similar way, consider another discrete stochastic process $\boldsymbol{U} = (U_1, \ldots, U_N)$, each $U_n \in \Lambda = \{1, 2, 3, 4\}$. The four values in Λ respectively represent stance, push-up, swing and step down. The number N denotes the length of the stochastic process[2] and the total number of possible combinations of states in X_n and U_n is $m = 4 \times 4 = 16$. Then, let $\boldsymbol{Y} = (\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_N)$ be a real-valued process representing the observation of the model, each $\boldsymbol{Y}_n \in \mathbb{R}^q$, where q is the observation dimension.

[2]Note that N is required for the model presentation but it will not be used for the on-line algorithm.



(a) Walking. (b) Running.



(c) Stair ascent. (d) Stair descent.

Figure 3: Angular rates of four activities for each sensor axis ($\omega^{\mathcal{X}}$, $\omega^{\mathcal{Y}}$, $\omega^{\mathcal{Z}}$). The signals show that the patterns of the four kinds of gait cycle are quite different.

Let now consider the process $\boldsymbol{T} = (\boldsymbol{X}, \boldsymbol{U}, \boldsymbol{Y})$. $\boldsymbol{T}$ is said to be a triplet Markov chain (TMC) if it is Markovian [12]. TMCs are strictly more general than pairwise Markov chains (PMCs) [23], which are themselves more general than HMMs, see for example [24] for detailed explanations. In a general TMC, none of the processes $\boldsymbol{X}, \boldsymbol{Y}, (\boldsymbol{X}, \boldsymbol{U}), (\boldsymbol{X}, \boldsymbol{Y}), (\boldsymbol{U}, \boldsymbol{Y})$ are necessarily Markovian [25], but the conventional algorithms of the parameter learning still work for TMCs. The BaumWelch algorithm (but not necessarily the Viterbi algorithm) applies in triplet Markov models, so Bayesian MPM (Maximum Posterior Mode) criterion can be used to recover both $\boldsymbol{X}$ and $\boldsymbol{U}$, *i.e.* activities and gait phases, from $\boldsymbol{Y}$ only, once parameters of the model are known. When assuming stationarity of processes, such classification algorithm can be made unsupervised, by estimating model parameters using the Expectation-Maximization (EM) principle under Gaussian distributions assumption [26]. Unfortunately, Gaussian assumption is not suited for our application, so we develop a non-parametric representation of data-driven densities. Meanwhile, the on-line algorithm presented in the next Section is designed to compensate for the stationary assumption.

Let the realizations of X_n, U_n and $\boldsymbol{Y}_n$ be denoted by x_n, u_n and $\boldsymbol{y}_n$ respectively, so $\boldsymbol{v}_n =$

(x_n, u_n), $\boldsymbol{t}_n = (\boldsymbol{v}_n, \boldsymbol{y}_n)$. Also, for simplification, we will denote the probability $p(X_n = x_n, U_n = u_n | \boldsymbol{Y}_1 = \boldsymbol{y}_1, \ldots, \boldsymbol{Y}_N = \boldsymbol{y}_N)$ by $p(x_n, u_n | \boldsymbol{y}_1^N)$ for example. The dependency graph of the specific TMC suited for our application is shown in Fig. 4. The transition probability of $\boldsymbol{T}$, $p\left(\boldsymbol{t}_{n+1} | \boldsymbol{t}_n\right) = p\left(\boldsymbol{v}_{n+1}, \boldsymbol{y}_{n+1} | \boldsymbol{v}_n, \boldsymbol{y}_n\right)$, is simplified to

$$p\left(\boldsymbol{t}_{n+1} | \boldsymbol{t}_n\right) = p\left(x_{n+1}, u_{n+1} | x_n, u_n\right) p\left(\boldsymbol{y}_{n+1} | x_{n+1}, u_{n+1}\right), \tag{1}$$

which provides to process $\boldsymbol{T} = (\boldsymbol{V}, \boldsymbol{Y})$, with $\boldsymbol{V} = (\boldsymbol{X}, \boldsymbol{U})$, the structure of a classical HMC. The first term $p\left(x_{n+1}, u_{n+1} | x_n, u_n\right)$ is the state transition probability, the dimension of the transition matrix being $m \times m$ (recalling that $m = 16$). For activity recognition, the state can only transfer from one gait phase to the next gait phase within the same activity, or, from stance phase of one activity to push-up phase of another activity, as shown in Fig. 5. Therefore, there are only 44 non-zero entries in the $m \times m$ joint probability matrix $p(\boldsymbol{v}_n, \boldsymbol{v}_{n+1})$. Process $\boldsymbol{V}$ has the shape of a cyclic left-right Markov chain. The second term in eq. (1) is the probability of observing $\boldsymbol{y}_n$ conditionally to each state. Most of the time, this kind of density is modeled by using Gaussian distributions, whereas in this work we propose a non-parametric modeling method by using histograms to represent $p(\boldsymbol{y}_n | \boldsymbol{v}_n)$ for its best ability to fit users' steps, whatever their size. The number of histograms to be managed is $m \times q$. The resulting model is called TMC-HIST in the remaining.

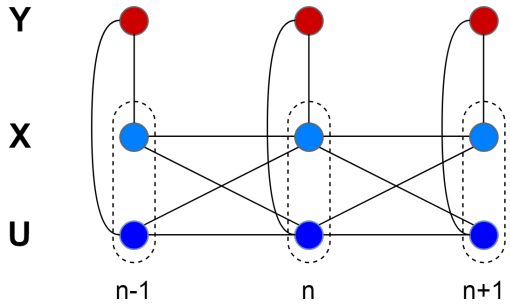


Figure 4: TMC dependency graph for activity recognition.

Let us now detail the Baum-Welch algorithm (also called "forward-backward procedure") suited to the TMC-HIST model. We are trying to calculate $p(x_n, u_n | \boldsymbol{y}_1^N)$. First, consider the forward and backward probabilities $\alpha_n(\boldsymbol{v}_n) = p(\boldsymbol{v}_n, \boldsymbol{y}_1^n)$ and $\beta_n(\boldsymbol{v}_n) = p(\boldsymbol{y}_{n+1}^N | \boldsymbol{v}_n)$, where $\boldsymbol{v}_n = x_n \times u_n \in \Psi \times \Lambda$.

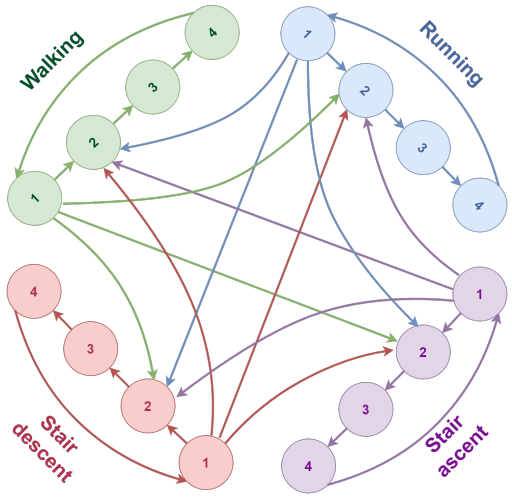


Figure 5: State transition graph of the TMC-based activity recognition algorithm. The values (1,2,3,4) represent the stance, push-up, swing and step down respectively, for the four gait phases.

Classically, they can be computed by the following two recursions, using eq. (1)

$$\begin{aligned}
\alpha_1(\boldsymbol{v}_1) &= p\left(\boldsymbol{v}_1, \boldsymbol{y}_1\right), \\
\alpha_n(\boldsymbol{v}_n) &= \sum_{\boldsymbol{v}_{n-1} \in \Psi \times \Lambda} \alpha_{n-1}(\boldsymbol{v}_{n-1})\, p\left(\boldsymbol{v}_n | \boldsymbol{v}_{n-1}\right) p\left(\boldsymbol{y}_n | \boldsymbol{v}_n\right), \\
\beta_N(\boldsymbol{v}_N) &= 1, \\
\beta_n(\boldsymbol{v}_n) &= \sum_{\boldsymbol{v}_{n+1} \in \Psi \times \Lambda} p\left(\boldsymbol{v}_{n+1} | \boldsymbol{v}_n\right)\; p\left(\boldsymbol{y}_{n+1} | \boldsymbol{v}_{n+1}\right) \beta_{n+1}(\boldsymbol{v}_{n+1}).
\end{aligned} \tag{2}$$

Then, $p(\boldsymbol{v}_n|\boldsymbol{y}_1^n)$, $p(\boldsymbol{v}_n|\boldsymbol{y}_1^N)$ and $p(\boldsymbol{v}_n, \boldsymbol{v}_{n+1}|\boldsymbol{y}_1^N)$ are computed by equations

$$p\left(\boldsymbol{v}_n | \boldsymbol{y}_1^n\right) = \frac{\alpha_n(\boldsymbol{v}_n)}{\sum\limits_{\boldsymbol{v}_n \in \Psi \times \Lambda} \alpha_n(\boldsymbol{v}_n)}, \tag{3}$$

$$p\left(\boldsymbol{v}_n | \boldsymbol{y}_1^N\right) = \frac{\alpha_n(\boldsymbol{v}_n)\beta_n(\boldsymbol{v}_n)}{\sum\limits_{\boldsymbol{v}_n \in \Psi \times \Lambda} \alpha_n(\boldsymbol{v}_n)\beta_n(\boldsymbol{v}_n)}, \tag{4}$$

$$p\left(\boldsymbol{v}_n, \boldsymbol{v}_{n+1} | \boldsymbol{y}_1^N\right) = \frac{\alpha_n(\boldsymbol{v}_n)\; p\left(\boldsymbol{v}_{n+1} | \boldsymbol{v}_n\right) p\left(\boldsymbol{y}_{n+1} | \boldsymbol{v}_{n+1}\right)\; \beta_{n+1}(\boldsymbol{v}_{n+1})}{\sum\limits_{\boldsymbol{v}_n, \boldsymbol{v}_{n+1} \in \Psi \times \Lambda} \alpha_n(\boldsymbol{v}_n)\; p\left(\boldsymbol{v}_{n+1} | \boldsymbol{v}_n\right) p\left(\boldsymbol{y}_{n+1} | \boldsymbol{v}_{n+1}\right)\; \beta_{n+1}(\boldsymbol{v}_{n+1})}. \tag{5}$$

The optimal classification $\hat{\boldsymbol{v}}_1^N = \{\hat{\boldsymbol{v}}_{\mathbf{1}}, \cdots, \hat{\boldsymbol{v}}_{\boldsymbol{N}}\}$, according to the Bayesian MPM criterion, is then given by

$$\hat{\boldsymbol{v}}_n = arg \max_{\boldsymbol{v}_n \in \Psi \times \Lambda} p\left(\boldsymbol{v}_n | \boldsymbol{y}_1^N\right), \tag{6}$$

for $n = 1, \ldots, N$.

These calculations can be performed once the parameters of the model, *i.e.* the $m \times m$ joint probability matrix $p\left(x_{n+1}, u_{n+1}, x_n, u_n\right)$ and the $m \times q$ histograms, are known. In the absence of learning data, the well-known Expectation-Maximization (EM) principle is generally applied for learning parameters because it provides the exact re-estimation formula for parameters under Gaussian distribution assumptions. As we deal with non-parametric histograms, we make use of another unsupervised learning method called "Iterative Conditional Estimation" (ICE) [23, 27], which is applicable in a wide range of situations. Here, we simply recall the ICE procedure:

(1) Initialize parameters by some method. This point will be discussed in details in the next section.

(2) Compute the forward-backward algorithm using current parameters, and compute the state transition probability conditioned on observations:

$$p\left(\boldsymbol{v}_{n+1} | \boldsymbol{v}_n, \boldsymbol{y}_1^N\right) = \frac{p\left(\boldsymbol{v}_n, \boldsymbol{v}_{n+1} | \boldsymbol{y}_1^N\right)}{p\left(\boldsymbol{v}_n | \boldsymbol{y}_1^N\right)}; \tag{7}$$

(3) Simulate a realization of state sequence $\tilde{\boldsymbol{v}} = (\tilde{\boldsymbol{v}}_1, \ldots, \tilde{\boldsymbol{v}}_N)$ by using $p(v_1 | \boldsymbol{y}_1^N)$ and $p(\boldsymbol{v}_{n+1} | \boldsymbol{v}_n, \boldsymbol{y}_1^N)$ given in eq. (4) and (7);

(4) Update the joint probability matrix and histograms according to $\tilde{\boldsymbol{v}}$ and $\boldsymbol{y}_1^N$;

(5) Stop when the number of ICE iterations reaches a predefined maximum value, 100 for example, else go back to step (2).

This unsupervised algorithm can only be applied in an off-line scenario, *i.e.* when all the data have been collected. Next section is devoted to present an algorithm suited for on-line data.

3. Adaptive On-line Classification Algorithm

The diagram of the adaptive on-line algorithm for lower limb locomotion activities classification is displayed in Fig. 6. The entire algorithm is composed of four stages: (i) model training, (ii) data acquisition and pre-processing, (iii) complete gait detection, (iv) final decision and posterior

update. The (i) stage will be described firstly, (ii) and (iii) stages will then be presented together, (iv) stage will finally be described.

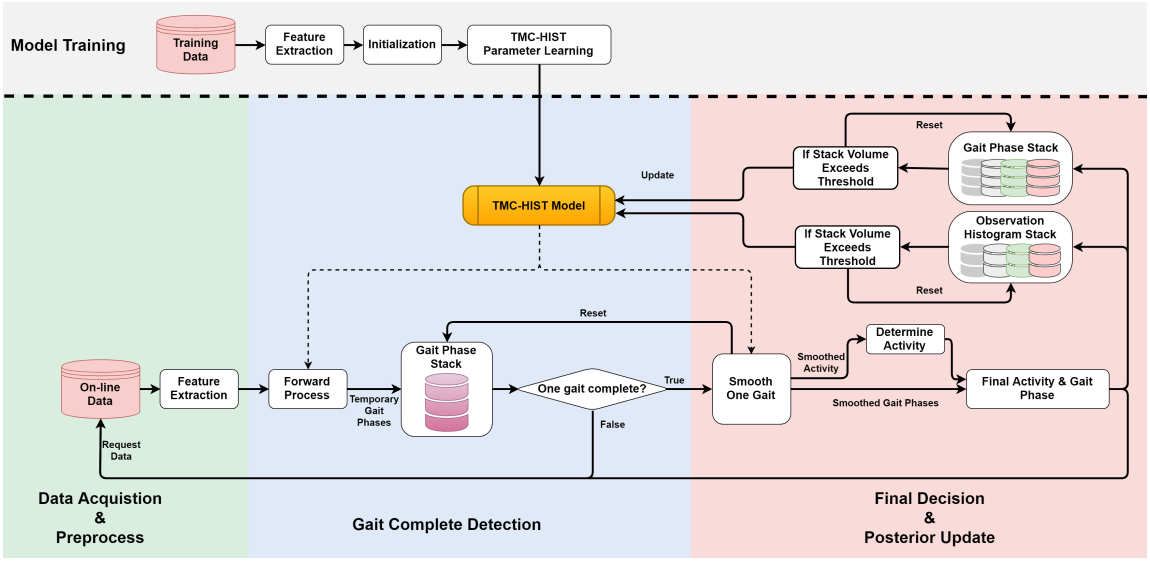


Figure 6: Diagram of the adaptive on-line inference algorithm.

3.1. Model Training

The TMC-HIST model needs to be trained before being used on-line, this trained model will act as an initial model for the on-line recognition. It guarantees that the algorithm is initialized in a status that will allow a roughly correct classification. The trained model needs to detect all kinds of activities and activity switches, thus the training data should contain the four activities and all possible activity switches. While it should be noticed that on-line data does not have this restriction.

Following [16] and according to our own tests, two kinds of features are selected for observations: the sliding mean μ_n and sliding standard deviation σ_n of the angular rates ω_j for the three axes of the sensor, computed at time n on a time period of length W:

$$\begin{aligned} \mu_n^i &= \frac{1}{W} \sum_{j=n-W+1}^{n} \omega_j^i, \\ \sigma_n^i &= \sqrt{\frac{1}{W} \sum_{j=n-W+1}^{n} \left(\omega_j^i - \mu_n^i\right)^2}, \end{aligned} \tag{8}$$

where $i \in \{\mathcal{X}, \mathcal{Y}, \mathcal{Z}\}$ represents the axis of sensor.

Hence, an observation $\boldsymbol{y}_n$ are composed of three mean values and three standard deviations, *i.e.* a $q = 6$-dimensional vector, and the total number of histograms in TMC-HIST is therefore 96. The three axes of the sensor are correlated in the movement, while the applied histograms represent the marginal probability density of each axis. Using a 3-dimensional histogram would probably suit the application more closely. But it would need too much data to form a reliable density and drastically increase the computation time.

Parameters learning needs $p(\boldsymbol{v}_n, \boldsymbol{v}_{n+1})$ and $p(\boldsymbol{y}_n|\boldsymbol{v}_n)$ to be initialized. The ground truth of activity x_n can be easily obtained from the sequence of activities used in the experiment. But unfortunately, the ground truth of gait phases u_n is unavailable because of the lacking of proper device to acquire it. Therefore, a semi-supervised initialization method is developed according to the following procedure:

(1) Set filtering cut-off frequency f_k and stance threshold h_k for each activity $k \in \Psi$. The value of f_k for each activity is set to $5Hz$, $9Hz$, $4.5Hz$, $6Hz$, respectively. The value of h_k for each activity is set to $0.52rad/s$, $1.92rad/s$, $0.52rad/s$, $0.52rad/s$, respectively. All these values were obtained through tests.

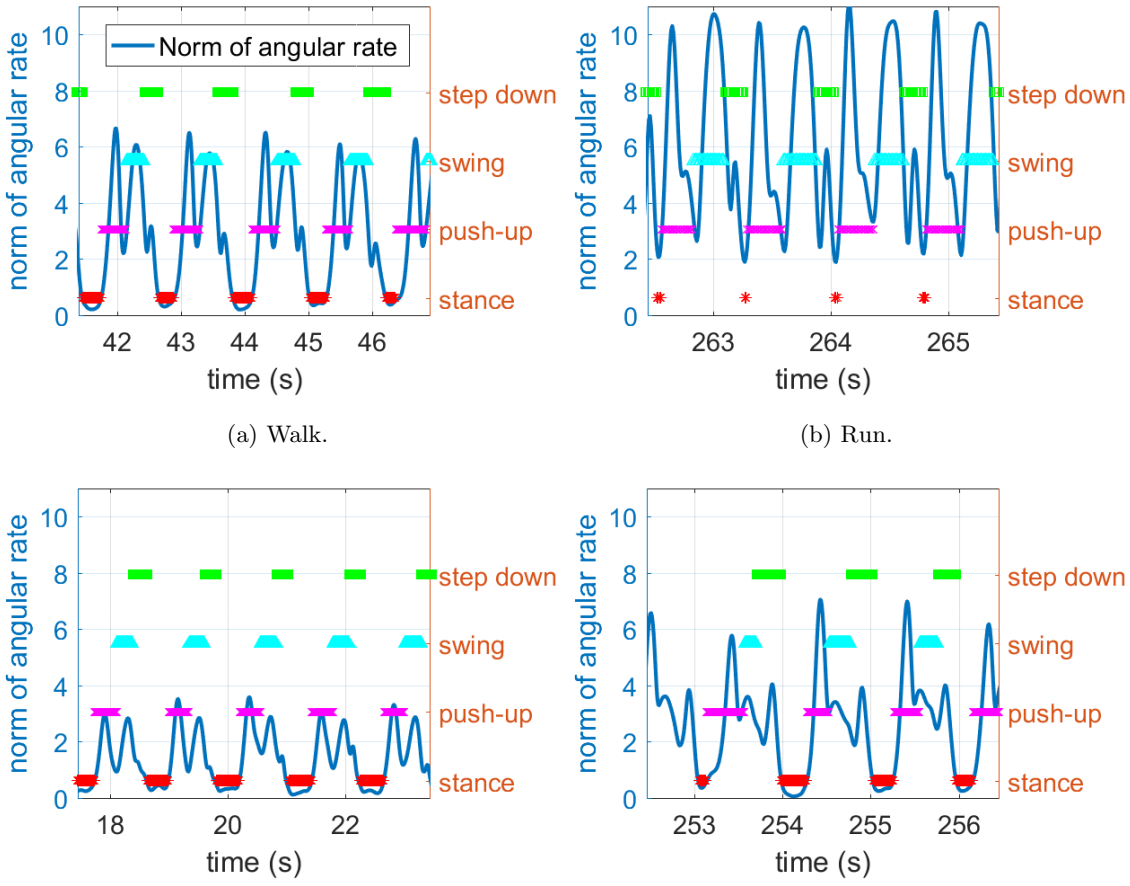
(2) Use a Butterworth low-pass filter to filter the norm of angular rate according to f_k, then segment the filtered angular rate by corresponding h_k. All the periods below the threshold will be regarded as stance gait phase, all the periods above will be regarded as non-stance phases.

(3) In each period of non-stance phase, the three peaks represent the three non-stance gait phases, *i.e.* push-up, swing, and step-down. Thus, the three non-stance gait phases can be initialized according to the peaks. Here, we simply use four indices: the start and end indices of the non-stance phase, the middle index between the first and second peaks, the middle index between the second and third peaks, to obtain the three non-stance gait phases.

(4) Once gait phases have been segmented for all the activities, $p(\boldsymbol{v}_n, \boldsymbol{v}_{n+1})$ and $p(\boldsymbol{y}_n|\boldsymbol{v}_n)$ can be easily obtained by using trivial empirical estimators.

An excerpt of the initialization procedure is given in Fig. 7.

Segmentation results are only roughly segmented by a very simple method and the initialized parameters are far from the optimal ones. Then ICE is applied to improve fitting parameters. One



(a) Walk.

(b) Run.

(c) Stair ascent.

(d) Stair descent.

Figure 7: In each sub-figures, blue line is the filtered angular rate norm. The red, purple, cyan and green represent stance, push-up, swing and step-down, respectively.

point should be clarified here: the training stage is a semi-supervised method since the ground truth of activity is only used for initialization. While the learning stage by ICE is an unsupervised method to get the final training model.

3.2. On-line Data Acquisition and Complete Gait Detection

The data is fed to the on-line algorithm once it arrives. Then feature extraction is performed based on the new data and the stored ones, by using eq. (8). The data is accumulated during the required time to obtain a complete gait with the sequence: push-up $\rightarrow$ swing $\rightarrow$ step down $\rightarrow$ stance.

In order to decide whether a gait is complete, a one-step forward process of TMC-HIST model is firstly conducted to estimate current gait phase, which derives from the marginal probability of $p(x_n, u_n|\boldsymbol{y}_1^n)$ over x_n, using MPM method

$$\tilde{u}_n = arg\max_{u_n \in \Lambda} \Big\{ \sum_{x_n \in \Psi} p\left(x_n, u_n|\boldsymbol{y}_1^n\right) \Big\}. \tag{9}$$

Then, the current gait phase and the features are stored in a stack, which contains all the gait phases and corresponding features of the current gait cycle. Afterward, a decision is made to verify whether an entire gait is complete or not. If the stack is shorter than a threshold t_{min} or does not contain a complete sequence of gait phases, the algorithm will return and wait for another new data to come in. The current forward result $p(x_n, u_n|\boldsymbol{y}_1^n)$, obtained from eq. (3), will be stored for the use of the forward process of the next sampling time. The value of t_{min} is set to $0.5s$ based on the fastest speed of gait cycle among the four activities. Otherwise, it means that an entire gait is completed and all the features stored will be sent to the last stage.

3.3. Final Decision and Posterior Update

The activity and gait phases in one detected gait cycle are denoted by x'_n and u'_n, and N' is the number of samples for the corresponding gait cycle. This stage proceeds only when one gait is completed and contains two steps: 1) final decision for activity and gait phases, 2) posterior update of the joint probability $p(\boldsymbol{v}'_n, \boldsymbol{v}'_{n+1}|\boldsymbol{y}_1^{N'})$ and the histograms that represent $p(\boldsymbol{y}_n|x_n, u_n)$, if necessary. First, the features from the previous stage are smoothed by the forward-backward process to get $p(\boldsymbol{v}'_n|\boldsymbol{y}_1^{N'})$, then the smoothed activities and gait phases are obtained using MPM criterion again:

$$\begin{aligned} \widehat{x'_n} &= arg\max_{x'_n \in \Psi} \Big\{ \sum_{u'_n \in \Lambda} p\left(x'_n, u'_n|\boldsymbol{y}_1^{N'}\right) \Big\}, \\ \widehat{u'_n} &= arg\max_{u'_n \in \Lambda} \Big\{ \sum_{x'_n \in \Psi} p\left(x'_n, u'_n|\boldsymbol{y}_1^{N'}\right) \Big\}. \end{aligned} \tag{10}$$

The final decision of gait phases is $\widehat{u'_n}$, for each sampling time. However, there is only one possible activity for each gait cycle, so the final decision of activity $\widehat{x'}$ depends on the most frequent estimated activity among $\widehat{x'_n}$:

$$\widehat{x'} = arg\max_{k \in \Psi} \Big\{ \sum_{n=1}^{N'} \mathbb{1}_{\widehat{x'_n}=k} \Big\} \tag{11}$$

where $\mathbb{1}$ is a boolean-valued function which takes value 1 when it satisfies the condition, or takes 0 if not. Then, all the $\widehat{x'_n}$ within one gait cycle are set to the final decision $\widehat{x'}$.

Based on the final decision of activity and gait phases, we can accumulate the features into the stacks for posterior update. In the gait phase stacks, there are four different 4×4 matrices for each activity and each matrix is a counter relative to the $p(u'_n, u'_{n+1}|\boldsymbol{y}_1^{N'})$, which is a marginal probability of $p(\boldsymbol{v}'_n, \boldsymbol{v}'_{n+1}|\boldsymbol{y}_1^{N'})$ over all x'_n and x'_{n+1}. The sequence of gait phases from final decision are divided into $N'-1$ pairs of (u'_n, u'_{n+1}), the number of each (u'_n, u'_{n+1}) case is then accumulated into the matrix according to $\widehat{x'}$. As the name suggests, the observation histograms stacks contain $m \times q$ different stacks and accumulate the observations, *i.e.* features, according to $\widehat{x'}$ and $\widehat{u'_n}$.

Two thresholds h_{gait} and $h_{observation}$ are applied here to decide whether these two kinds of stacks are large enough to update the TMC-HIST model. When one matrix in gait phase stacks has accumulated more than h_{gait} gaits, the joint probability of the relative activity will be calculated, and the corresponding joint probability in TMC-HIST model will be updated. After that, all the entries in this matrix are reset to zero. Likewise, the histograms in TMC-HIST model are replaced if the related stack's volume exceeds $h_{observation}$; reset is also conducted after the update. It should be noticed is that all the matrices and histogram stacks are accumulated and updated independently, because the duration of each gait phase is different from each other. Thanks to the posterior update, the on-line algorithm can adjust the parameters in TMC-HIST model according to the users' activity patterns and the road conditions, such as the speed of foot strike, ascent or descent slope...

4. Experimental Results

Ten healthy subjects were invited for the experiment: three females and seven males, age from 25 to 47 years old, weight from 47 to 83 kg, height from 160 to 184 cm. The sensor sampling rate was set to $100Hz$, and the range of gyroscope to $\pm 1000deg/s$. The window size W for feature extraction was set to 15 samples based on our experience, which corresponds to $0.15s$. It was determined by the stance duration when running, which is the shortest gait phase duration of all the activities. This is reasonable since a window size bigger than the duration may reduce the detection accuracy of the shortest gait phase, whereas a too small W may not be sufficient for calculating the mean value and standard deviation. The ranges of the histograms in TMC-HIST

were set from $-15rad/s$ to $15rad/s$ for mean value, and from $0rad/s$ to $15rad/s$ for standard deviation, and the bins number of the histograms was all set to 300.

The proposed algorithm is evaluated on a designed experiment, which is conducted by the subjects. We utilize a 2-fold cross-validation method, by equally separating the subjects into two groups. This way guarantees that training data and testing data come from different sources. The experiment was conducted around a building with four floors on the campus of École Centrale de Lyon (France). The selected path consists of walking and running around the building (with ramp road conditions), as well as climbing and descending stairs. The exact sequence of activities is

1. 600 meters of walking,
2. 600 meters of running,
3. four round trips of climbing stairs from ground floor to the 4th floor and back to the ground floor,
4. repeat steps 1. to 3. a second time.

Hence, one experiment consists in repeating two times the same sequence of activities, called here "a section". The ground truth of the experiment can be seen in Fig. 10. It consists of $1200m$ of walking and running, 32 floors of stair ascent and descent. The average time to complete one experiment is about 30 minutes. All the subjects perform the activities at their preferred speed, but are asked to keep the same speed within one experiment as much as possible. This ensures that the activity patterns and speeds vary among the subjects, but almost keep constant within one experiment. The speed ranges of the four activities among the subjects are $1.09 - 1.68m/s$, $2.14 - 3.82m/s$, $89.79 - 123.68$ stairs/min, $96.07 - 140.97$ stairs/min.

The implementation of the algorithms is done in Matlab, the code is running on a $3.2GHz$ CPU computer with 64-bit Win7 operating system. The average experiment time of all the subjects is 29.80 minutes, while the average calculation time by code is 2.60 minutes. Since calculation time are more than 11 times faster than that experimental one, thus the processing can be theoretically made on-line on a processor of frequency higher than $3.2GHz/11 = 291MHz$.

A global analysis over the ten subjects will be displayed first, to show a general performance of the batch mode classification and of the adaptive classification algorithm we propose. Then, one subject's data will be used for precisely analyzing the parameters updating.

The confusion matrix of batch mode classification using TMC-HIST is shown in Table 1. The overall accuracy is 83.17% and the Matthews correlation coefficient (MCC[3] [28]) is 0.7823. For the four activities, namely walking, running, stair ascent and descent, the sensitivities are 0.7434, 0.9538, 0.8570, 0.9215, respectively; the specificities are 0.9715, 0.8834, 0.9610, 0.9656, respectively.

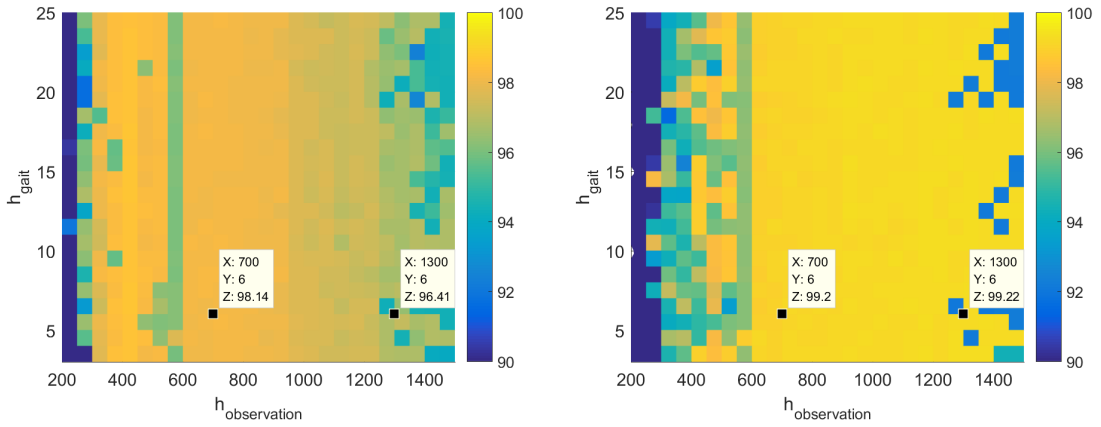
Table 1: Confusion matrix of batch mode classification.

		Predicted activity			
		Walk	**Run**	**Stair ascent**	**Stair descent**
True activity	**Walk**	**74.34%**	16.71%	5.45%	3.50%
	Run	3.79%	**95.38%**	0.54%	0.29%
	Stair ascent	2.45%	4.00%	**85.70%**	7.85%
	Stair descent	1.82%	2.32%	3.70%	**92.15%**

For investigating the influences of h_{gait} and $h_{observation}$ –the size of the stacks before updating TMC-HIST– have on the activity classification performance. Fig. 8 shows the influence on average accuracy over the ten subjects *w.r.t.* different values of the thresholds h_{gait} and $h_{observation}$. From the results, $h_{observation}$ affects the accuracy more than h_{gait} does, which means that the update of histograms is more critical than the update of joint probability matrix. The accuracy obtained in the second section is higher than in the first section, which indicates our proposed on-line algorithm can adjust the parameters in TMC-HIST model properly and improve the classifying accuracy gradually. We can see that, as the value of $h_{observation}$ increases, the average accuracy reaches above 99% when $h_{observation} \in [600, 750]$. According to this experiment, we select $h_{observation} = 700$ and $h_{gait} = 6$ to analyse the performance of the proposed on-line algorithm.

Confusion matrices of the first and second sections of the experiment are shown in Table 2. We can see that the classification performance of each activity is improved from the first to the second. The sensitivities of the four activities (walking, running, stair ascent and descent) are 0.9869, 0.9944, 0.9765, 0.9470 respectively in the first section, while they are 0.9939, 0.9983, 0.9862, 0.9820 in the

[3]MCC is a measure for multi-category classification, it can balance the influence that produced by the different proportion of each category, a value close to 1 means a perfect classification.



(a) (b)

Figure 8: Averaged activity recognition accuracy (in %) according to $h_{observation}$ and h_{gait} (size of stacks), for the first section of experiment (a) and for the second section (b).

second section. Similarly, the specificities are 0.9968, 0.9940, 0.9897, 0.9959 in the first section; while they are 0.9965, 0.9987, 0.9958, 0.9982 in the second section. The overall accuracy increases from 98.14% to 99.20%, and MCC increases from 0.9682 to 0.9869. The better performances in the second section show that the parameters adjustment in the proposed on-line algorithm is beneficial, which gradually leads to an improvement of classification performances.

In order to intuitively understand the adaptation of the proposed on-line algorithm, Fig. 9 shows the accuracy of each activity in the most recent 1000 samples, *i.e.*, the accuracy that calculated from the latest 10 seconds with respect to each activity. The most recent accuracy can be used for investigating the converging rate. As shown in the figure, running is the fastest to converge at an accuracy over 99%, walking converges at about $40s$, stair ascent and descent reach a relatively high accuracy after $100s$. The drops in accuracy after convergence is due to the activity switch, which may affect the classification behavior during a short time, but will perform well again after the activity switch. It is not difficult to find that walking and running are easier to be classified than climbing stairs, showing that the model approximation with respect to climbing stairs is slower than that of walking and running. Nevertheless, all the activities reach a high accuracy level in the experiments finally.

For understanding parameters updating, a typical subject's data is selected here (subject 5,

male, 25 years old, 52kg weight and 170cm tall). One reason for selecting this experimental data is because the accuracy and MCC in the second section, 98.99% and 0.9800 respectively, are lower than the averaged values of all the subjects. So, Fig. 10 shows the ground truth and classified activities of the subject 5. It can be seen that, in the second section, some samples of walking are wrongly classified as stair descent. And stair ascent and stair descent are misclassified at the begin-

Table 2: Confusion matrix of the first section (up) and second section (down) of experiments.

		Predicted activity			
		Walk	**Run**	**Stair ascent**	**Stair descent**
True activity	**Walk**	**98.69%**	0.58%	0.37%	0.36%
	Run	0.48%	**99.44%**	0.00%	0.08%
	Stair ascent	0.13%	1.18%	**97.65%**	1.05%
	Stair descent	0.29%	0.05%	4.96%	**94.70%**

		Predicted activity			
		Walk	**Run**	**Stair ascent**	**Stair descent**
True activity	**Walk**	**99.39%**	0.21%	0.23%	0.16%
	Run	0.16%	**99.83%**	0.01%	0.00%
	Stair ascent	0.86%	0.00%	**98.62%**	0.52%
	Stair descent	0.09%	0.00%	1.71%	**98.20%**

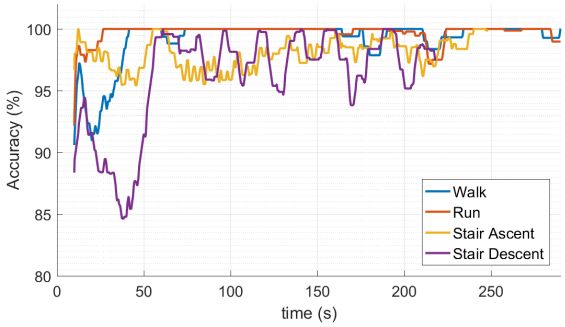


Figure 9: Accuracy in the most recent 1000 samples *w.r.t.* each activities.

ning. But as expected, the performance in the second section is much better than in the previous one. Regarding the gait phases, there is no precise evaluation method because of the lacking of the proper device to collect the ground truth of gait cycles. However, as shown in Fig. 11, it is obvious that the gait cycles detection becomes more regular when the model's parameters have converged to fit the subject gait rhythm. Indeed, gait cycles are introduced in the proposed algorithm to make the on-line algorithm possible and help to obtain a better activity classifying result, therefore, a well-detected gait cycle assists to obtain a good classification of activity.

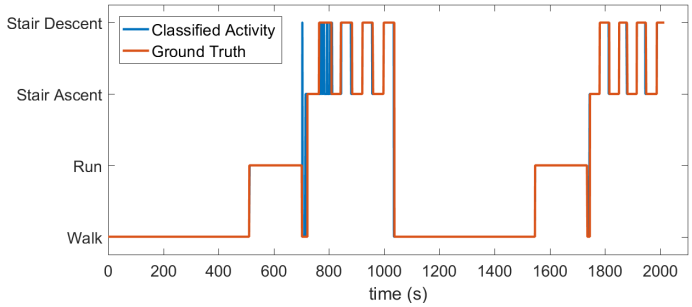


Figure 10: Classified activities from the proposed algorithm for subject 5. For the first section, accuracy and Matthews correlation coefficient are 94.49% and 0.9055 respectively, while, for the second section, the values are 98.99% and 0.9800 respectively.

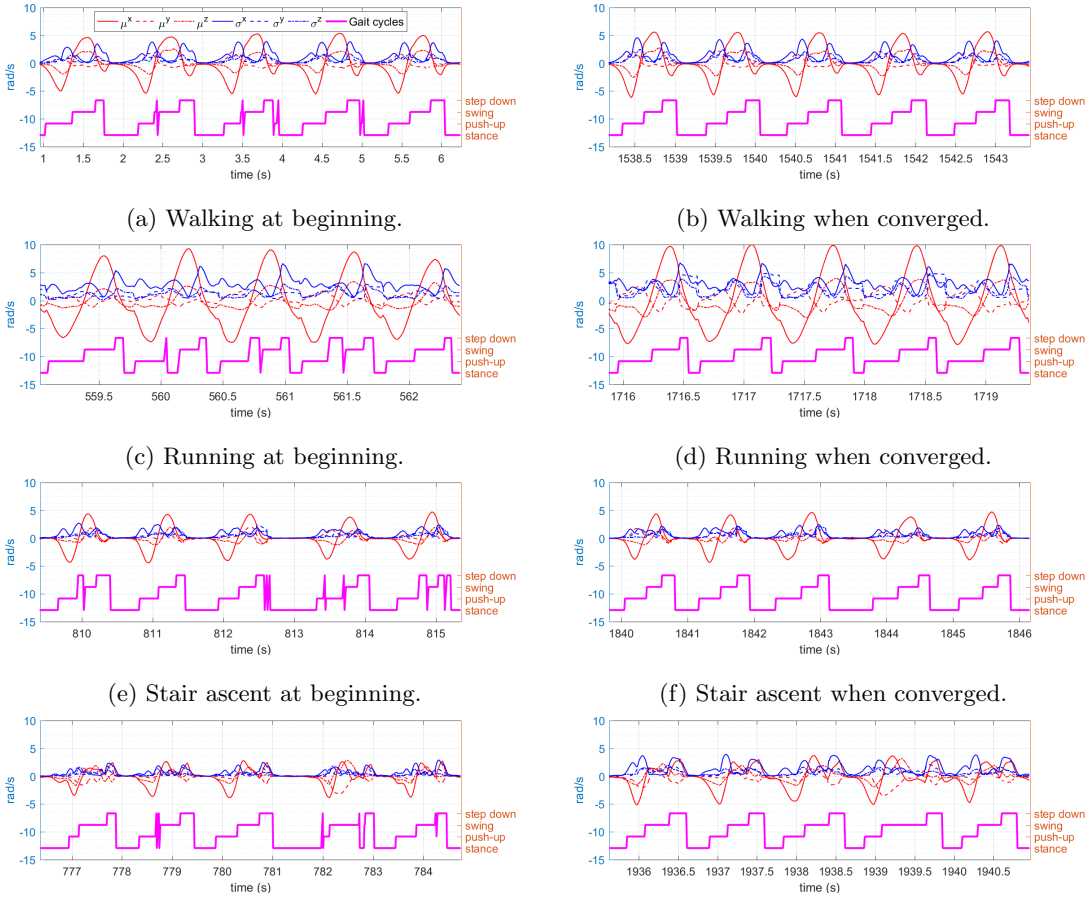
5. Discussion

This section proposes an in-depth discussion on the experimental results for key points of the proposed algorithm, such as the classification accuracy, convergence rate, gait phase detection, parameter update...

5.1. Classification Performance

A comparison of the performances of the proposed adaptive on-line algorithm compared to some state-of-the-art algorithms is displayed in Table 3. These algorithms are evaluated according to several aspects including the number of sensors, the average accuracy, the MCC...

Among all the works that use one single sensor reported in Table 3, the proposed algorithm obtains the highest average accuracy with 99.20%. While the Phase Variable obtained an accuracy of 98.30% for only 3 activities: walking, stair ascent and descent. Descriptor-based methods obtained



(a) Walking at beginning.

(b) Walking when converged.

(c) Running at beginning.

(d) Running when converged.

(e) Stair ascent at beginning.

(f) Stair ascent when converged.

(g) Stair descent at beginning.

(h) Stair descent when converged.

Figure 11: The detected gait cycles at the beginning of each activity (left column) compared to the ones when the estimation of the model's parameter has converged (right column).

97.12% on 6 lower limb activities, whereas energy expenditure prediction obtained 95.05% on 5 lower limb activities. Adaboost Stump got an accuracy range from 95%-98% on 5 lower limb activities and PLP+HMM obtained $97.5 \pm 1.6\%$ on 6 activities, with 3 lower limb activities identical to the work of Phase Variable. In the group of multiple sensors, Log-Sum distance used 5 sensors to classify 6 activities (standing still, sitting still, laying down, walking forward, stair ascent and descent). The accuracy of Log-sum distance can reach 99%. Empirical mode decomposition used 17 sensors to classify 9 activities and obtained an overall accuracy at 97.78%. Among the 9 activities, the 7

Table 3: Comparison of the performance *w.r.t.* state-of-the-art algorithms for classifying lower limb locomotion activities.

Method	**Sensors**	**Position**	**No. of activity/ subject**	**Accuracy**	**MCC**	**On-line**
Adaptive BasIS (Martinez-Hernandez, 2018 [11])	3	foot, shank thigh	3/8	99.87%	—	NO
Phase Variable (Harrison & Michael, 2018, [29])	1	thigh	3/7	98.30%	—	NO
NWFE+PCA+LS-SVM (Yu-Liang Hsu, 2018, [30])	2	ankle, wrist	10/23	99.65% (daily activity)	0.9804	NO
Descriptor-based (Ankita Jain, 2018, [31])	1	waist	6/30	97.12%	0.966	NO
Log-Sum Distance (Faisal Sikder, 2017, [32])	5	waist, wrists ankles	6/3	99.00%	0.9734	NO
Energy expenditure prediction (Lin, 2017, [33])	1	foot	5/10	95.05%	0.9336	NO
Adaboost Stump (Li, 2017, [34])	1	trouser pocket	5/12	$95\% - 98\%$	—	YES
Empirical mode decomposition (Ayachi, 2016, [35])	17	whole body	9/4	97.78%	0.823	NO
PLP+HMM (San-Segundo, 2016, [8])	1	waist	6/30	$97.5 \pm 1.6\%$	—	NO
The proposed algorithm (TMC-HIST)	1	foot	4/10	99.20%	0.9869	YES

lower limb activities had an accuracy at about 96%. Adaptive BasIS used 3 sensors and obtained 99.87% on three activities: walking on level ground, ramp ascent and descent. NWFE+PCA+LS-SVM used 2 sensors and obtained an overall accuracy at 99.65% for 10 daily activities. Among the 10 activities, the 5 lower limb activities had an accuracy at about 96.46%. If looking at all the algorithms that give an accuracy higher than 99%, the proposed algorithm is the only one that uses one single sensor, besides, the proposed algorithm obtained the highest MCC value. The use of multiple sensors is very interesting and can give very high accuracy. Nevertheless, recognizing

activities with only one sensor is still relevant because it is more realistic than multiple sensors for quantified-self applications. It should be noticed that the state-of-the-art works and our algorithm are tested on different datasets, consisting of different activities and different amounts of samples. But from a general view of the results, we can still state that our proposed algorithm obtain performances comparable to or even better than the best ones, whereas allowing on-line processing.

5.2. Convergence Rate

It can be deduced from Fig. 8 and 9 that the convergence rate is affected by two aspects: the thresholds ($h_{observation}$, h_{gait}) and the activity category. These two aspects will be discussed in the following.

The first aspect is the thresholds that control the parameters updating, *cf.* Fig. 8. In the first experimental section, the accuracy at $h_{observation} = 1300$ is relatively lower than that of $h_{observation} = 700$ (96.41% compared to 98.14%), which means that using a small value of $h_{observation}$ gives a faster convergence rate than using a large value. While in the second section, the accuracies where $h_{observation} = 700$ and $h_{observation} = 1300$ are improved to 99.2% and 99.22% respectively. The close performance in the second section indicates that high accuracy can be reached through parameter updating with sufficient data. It should be noticed that in Fig. 8b , there are some isolated values when $h_{observation}$ is larger than 1200. This is because that in the experiment of one special subject, a large quantity of walking data are classified as running at the beginning, and the too late updating makes all the consequent walking data be classified as running. On the contrary, we can see that a too small $h_{observation}$ leads to a worse result in the second section compared to the first section. It is due to the fact that the accumulated data number in each stack is not enough for representing the distribution of $p(\boldsymbol{y}_n|\boldsymbol{v}_n)$. A large number of bins in histogram requires more data to form the proper density, if $h_{observation}$ is too small, the updated histograms can not properly represent $p(\boldsymbol{y}_n|\boldsymbol{v}_n)$. In this case, the classification performance will be not too bad since the initial model is obtained from training data. But, after several updates by the inadequate observation stacks, TMC-HIST model will be far from the actual one, which results in a reduction of accuracy. Therefore, based on the accuracy shown in Fig. 8b, we recommend $h_{observation}$ to be equal to $2-2.5$ times the bins number of histograms.

It seems like the variation in h_{gait} has little impact on the accuracy. The main reason is that the transition probability $p(\boldsymbol{t}_{n+1}|\boldsymbol{t}_n)$ where $\boldsymbol{v}_{n+1} \neq \boldsymbol{v}_n$ is very small for all kinds of $\boldsymbol{v}_n$ and all subjects.

Indeed, each state will keep the same for a period of time and then transfer to the next state. Thus, a slight change in $p(\boldsymbol{v}_n, \boldsymbol{v}_{n+1})$ will not lead to a significant change in the classification performance.

Another aspect is the activity category. Fig. 9 shows that walking and running are faster than the other two to reach an accuracy higher than 99%. This means that the parameters in TMC-HIST relative to walking and running converge faster than those of stair ascent and stair descent. This phenomenon is certainly to the amount of effort required to climb 32 floors. It is then more difficult to keep the same pace within one experiment. As shown in Fig. 11, we can find that the feature patterns of each gait cycle corresponding to walking and running are similar in both two experimental sections, while the distinctions of stairs ascent and descent are larger than the former two activities. As a result, convergence rates corresponding to stair ascent and descent are slowed down by the distinction, and that is also why the accuracy fluctuates extensively after reaching a high value. Regarding the fastest convergence rate (running), the signal values are much higher than the others, which indicates that the running signal is unique from the others and easy to distinguish.

5.3. Adaptation and On-line

As a matter of fact, the gait complete detection stage of the algorithm makes the on-line possible, while the posterior update stage enables the TMC-HIST model to be adaptive. Introducing the gait cycle to assist locomotion is practical, because each gait phase gives a prior condition to confine the number of possible states to be estimated in TMC-HIST. Indeed, the Adaptive BasIS algorithm [11] also introduced gait cycles to help classifying three activities of level ground walking, ramp ascent and descent walking. The Phase Variable algorithm tried to segment each gait cycle to classify the activities of level ground walking, stair ascent and descent activities. Both of these two algorithms obtained good results, as illustrated in section 5.1, but they did not utilize the actual structure of the gait phases in one gait cycle, *i.e.* the transition sequence of gait phases. While in our proposed algorithm, a Markov chain is established to mimic this structure. Generally, segmentation of gait cycles can only be accomplished in an off-line scenario, it is difficult to make a decision if only one sample is known. The authors of Phase Variable algorithm [29] stated that their method can be applied to the on-line scenario if using a small time window. However, thanks to the state transitions among the activities and gait phases, gait phase can be estimated at each sampling time with the help of previous estimation. Activity is then detected after the current gait cycle is

completed, and does not need to wait for the entire data or to set a time window for segmentation. It should be noticed that the gait detection is based on the forward procedure of TMC-HIST model, which is not as reliable as the backward procedure. A precise evaluation of the algorithm behavior with respect to gait phases was not conducted because of the lacking of proper device to collect ground truth.

Many adaptive methods were proposed for motion analysis. A machine learning method using adaptive local motion descriptor in [36] was proposed for recognizing human motion in videos. A fast and adaptive sparse representation method in [37] reached an accuracy up to 94% for the recognition of human activities using wearable sensors. Zhang [38] proposed an adaptive time window method for human activity classification and reached an accuracy up to 99.2%. Hameed [39] proposed adaptive zero crossing technique to detect muscle activity based on electrocardiography signals. Li [34] and Wen [17] used Adaboost for human activity classification with inertial sensors, and obtained accuracy up to 98%, particularly, Li's method was applied in on-line scenarios. In our proposed algorithm, the adaptive functionality is conducted by the posterior update stage, by updating the parameters in TMC-HIST to approximate the user's activity patterns. Combining with the gait cycle detection, our results show that a roughly estimated gait cycle makes posterior update works appropriately, then activity classification is improved because of parameters adaptation. See Fig. 11, for each activity, the detected gait cycles are more regular than the ones detected at the beginning. In return, the improved activity classification results can refine the gait cycle detection till to the gait cycle detection behaves well.

5.4. Limitations

The proposed algorithm is of interest in the fields that related to human activity recognition and gait analysis, especially for those lower limb activities that have a periodic pattern of feet or legs. But, we understand that there are limitations in the proposed algorithm. The presently proposed algorithm only works for motions, while static activities are not involved, such as standing, sitting, lying, *etc.* Another limitation is that the initial model cannot be too far away from the testing data, otherwise, the algorithm cannot obtain acceptable classification results for updating the parameters. In the same way, if one subject makes a huge shift in the pattern of one activity, it may probably cause a failure. However, based on the analysis of the data used in this paper, we can state that a shift does not exceed the range described in the section 4 may possibly work for

the proposed algorithm.

6. Conclusion and future work

We propose an adaptive and on-line algorithm for classifying lower limb locomotion activities, by using a foot-mounted IMU sensor. The activities include walking, running, stair ascent and descent, and share a similar gait phase sequence: stance $\rightarrow$ push-up $\rightarrow$ swing $\rightarrow$ step down. The key-idea is to detect the gait cycle by estimating gait phases, the activity is then classified through each gait cycle. This is done by a specific triplet Markov chain (TMC) model which allow the joint classification of gait phases and activity thanks to two discrete hidden processes. In particular, the originality of the TMC considered is the introduction of non parametric observation densities, adaptively estimated with histograms. Therefore, a TMC-HIST with a specific state transition graph is proposed to mimic the natural sequence of gait sequence and activity switches. Next, a TMC-HIST-based adaptive on-line algorithm is developed, which can automatically adjust the parameters in TMC-HIST to approximate subject's activity patterns from an initial model. The initial model is learned by an semi-supervised parameter learning method. Experimental results show that TMC-HIST is capable of estimating gait phases and activities accurately, and the adaptive on-line algorithm obtains a high performance in activity classification.

The future work will focus on the signals issued from the same kind of sensor but located in the pocket, which is more realistic than on a foot when considering a quantified-self scenario. The difficulty will then be to precisely detect gait phases and cycles since the sensor cannot directly measure the kinematic information of the foot. Another problem that needs to be solved is introducing more activities which do not contain gait cycles (standing, sitting or making turn) or different gait cycles (cycling). If the actual TMC model is not efficient in that situation, introducing semi-Markov model or switching model in TMC may be helpful.

Acknowledgement

We would like to thank the Chinese Scholarship Council (CSC) which partially supports this research.

References

[1] M. Amboni, L. Iuppariello, A. Iavarone, A. Fasano, R. Palladino, R. Rucco, M. Picillo, I. Lista, P. Varriale, C. Vitale, et al., Step length predicts executive dysfunction in Parkinsons disease: a 3-year prospective study, Journal of neurology 265 (10) (2018) 2211–2220.

[2] P. Sorrentino, A. Barbato, L. Del Gaudio, R. Rucco, P. Varriale, M. Sibilio, P. Strazzullo, G. Sorrentino, V. Agosti, Impaired gait kinematics in type 1 gauchers disease, Journal of Parkinson's disease 6 (1) (2016) 191–195.

[3] C.-Y. Chiang, K.-H. Chen, K.-C. Liu, S. Hsu, C.-T. Chan, Data collection and analysis using wearable sensors for monitoring knee range of motion after total knee arthroplasty, Sensors 17 (2) (2017) 418.

[4] S. Bao, S. Yin, H. Chen, W. Chen, A wearable multimode system with soft sensors for lower limb activity evaluation and rehabilitation, in: 2018 IEEE Int. Instrumentation and Measurement Technology Conf. (I2MTC), 2018, pp. 1–6. doi:10.1109/I2MTC.2018.8409880.

[5] M. Liparoti, M. Della Corte, R. Rucco, P. Sorrentino, M. Sparaco, R. Capuano, R. Minino, L. Lavorgna, V. Agosti, G. Sorrentino, et al., Gait abnormalities in minimally disabled people with multiple sclerosis: A 3D-motion analysis study, Multiple sclerosis and related disorders.

[6] M. M. Hassan, M. Z. Uddin, A. Mohamed, A. Almogren, A robust Human activity recognition system using smartphone sensors and deep learning, Future Generation Computer Systems 81 (2018) 307–313.

[7] M. Mobark, S. Chuprat, T. Mantoro, Improving the accuracy of complex activities recognition using accelerometer-embedded mobile phone classifiers, in: Second Int. Conf. on Informatics and Computing (ICIC), 2017, pp. 1–5. doi:10.1109/IAC.2017.8280606.

[8] R. San-Segundo, J.-M. Montero, R. Barra-Chicote, F. Fernández, J.-M. Pardo, Feature extraction from smartphone inertial signals for Human activity segmentation, Signal Processing 120 (2016) 359 – 372.

[9] N. Neverova, C. Wolf, G. Lacey, L. Fridman, D. Chandra, B. Barbello, G. Taylor, Learning Human identity from motion patterns, IEEE Access 4 (2016) 1810–1820.

[10] R. Rucco, A. Sorriso, M. Liparoti, G. Ferraioli, P. Sorrentino, M. Ambrosanio, F. Baselice, Type and location of wearable sensors for monitoring falls during static and dynamic tasks in healthy elderly: a review, Sensors 18 (5) (2018) 1613.

[11] U. Martinez-Hernandez, A. A. Dehghani-Sanij, Adaptive Bayesian inference system for recognition of walking activities and prediction of gait events using wearable sensors, Neural Networks 102 (2018) 107 – 119.

[12] W. Pieczynski, C. Hulard, T. Veit, Triplet Markov chains in hidden signal restoration, in: Image and Signal Processing for Remote Sensing VIII, Vol. 4885, Int. Society for Optics and Photonics, 2003, pp. 58–69.

[13] D. Wu, Z. Wang, Y. Chen, H. Zhao, Mixed-kernel based weighted extreme learning machine for inertial sensor based Human activity recognition with imbalanced dataset, Neurocomputing 190 (2016) 35–49.

[14] Z. Chen, Q. Zhu, Y. C. Soh, L. Zhang, Robust Human activity recognition using smartphone sensors via CT-PCA and online SVM, IEEE Transactions on Industrial Informatics 13 (6) (2017) 3070–3080.

[15] A. Moschetti, L. Fiorini, D. Esposito, P. Dario, F. Cavallo, Daily activity recognition with inertial ring and bracelet: An unsupervised approach, in: 2017 IEEE Int. Conf. on Robotics and Automation (ICRA), 2017, pp. 3250–3255. doi:10.1109/ICRA.2017.7989370.

[16] E. Fullerton, B. Heller, M. Munoz-Organero, Recognising Human activity in free-living using multiple body-worn accelerometers, IEEE Sensors Journal 17 (16) (2017) 5290–5297.

[17] J. Wen, Z. Wang, Sensor-based adaptive activity recognition with dynamically available sensors, Neurocomputing 218 (2016) 307–317.

[18] A. Ignatov, Real-time Human activity recognition from accelerometer data using convolutional neural networks, Applied Soft Computing 62 (2018) 915 – 922. doi:https://doi.org/10.1016/j.asoc.2017.09.027.
URL http://www.sciencedirect.com/science/article/pii/S1568494617305665

[19] J. Perry, J. R. Davids, Gait analysis: normal and pathological function, Journal of Pediatric Orthopaedics 12 (6) (1992) 815.

[20] N. Shetty, S. Bendall, Understanding the gait cycle, as it relates to the foot, Orthopaedics and Trauma 25 (4) (2011) 236 – 240. doi:https://doi.org/10.1016/j.mporth.2011.04.009.
URL http://www.sciencedirect.com/science/article/pii/S1877132711000625

[21] Z. Wang, J. Li, J. Wang, H. Zhao, S. Qiu, N. Yang, X. Shi, Inertial sensor-based analysis of equestrian sports between beginner and professional riders under different horse gaits, IEEE Transactions on Instrumentation and Measurement (99) (2018) 1–13.

[22] S. Qiu, Z. Wang, H. Zhao, H. Hu, Using distributed wearable sensors to measure and evaluate Human lower limb motions, IEEE Transactions on Instrumentation and Measurement 65 (4) (2016) 939–950.

[23] S. Derrode, W. Pieczynski, Signal and image segmentation using pairwise Markov chains, IEEE Transactions on Signal Processing 52 (9) (2004) 2477–2489. doi:10.1109/TSP.2004.832015.

[24] I. Gorynin, H. Gangloff, E. Monfrini, W. Pieczynski, Assessing the segmentation performance of pairwise and triplet Markov models, Signal Processing 145 (2018) 183 – 192. doi:https://doi.org/10.1016/j.sigpro.2017.12.006.
URL http://www.sciencedirect.com/science/article/pii/S0165168417304206

[25] P. Lanchantin, J. Lapuyade-Lahorgue, W. Pieczynski, Unsupervised segmentation of randomly switching data hidden with non-Gaussian correlated noise, Signal Processing 91 (2) (2011) 163–175.

[26] V. Némesin, S. Derrode, Robust blind pairwise Kalman algorithms using QR decompositions, IEEE Transactions on Signal Processing 61 (2013) 5–9. doi:10.1109/TSP.2012.2222383.

[27] J. P. Delmas, An equivalence of the EM and ICE algorithm for exponential family, IEEE Transactions on Signal Processing 45 (10) (1997) 2613–2615.

[28] J. Gorodkin, Comparing two K-category assignments by a K-category correlation coefficient, Computational biology and chemistry 28 (5-6) (2004) 367–374.

[29] H. L. Bartlett, M. Goldfarb, A phase variable approach for IMU-based locomotion activity recognition, IEEE Transactions on Biomedical Engineering 65 (6) (2018) 1330–1338. doi:10.1109/TBME.2017.2750139.

[30] Y. Hsu, S. Yang, H. Chang, H. Lai, Human daily and sport activity recognition using a wearable inertial sensor network, IEEE Access 6 (2018) 31715–31728. doi:10.1109/ACCESS.2018.2839766.

[31] A. Jain, V. Kanhangad, Human activity classification in smartphones using accelerometer and gyroscope sensors, IEEE Sensors Journal 18 (3) (2018) 1169–1177.

[32] F. Sikder, D. Sarkar, Log-sum distance measures and its application to Human-activity monitoring and recognition using data from motion sensors, IEEE Sensors Journal 17 (14) (2017) 4520–4533. doi:10.1109/JSEN.2017.2707921.

[33] S. Lin, Y. Lai, C. Hsia, P. Su, C. Chang, Validation of energy expenditure prediction models using real-time shoe-based motion detectors, IEEE Transactions on Biomedical Engineering 64 (9) (2017) 2152–2162.

[34] P. Li, Y. Wang, Y. Tian, T. Zhou, J. Li, An automatic user-adapted physical activity classification method using smartphones, IEEE Transactions on Biomedical Engineering 64 (3) (2017) 706–714.

[35] F. S. Ayachi, H. P. Nguyen, E. G. de Brugiere, P. Boissy, C. Duval, The use of empirical mode decomposition-based algorithm and inertial measurement units to auto-detect daily living activities of healthy adults, IEEE Transactions on Neural Systems and Rehabilitation Engineering 24 (10) (2016) 1060–1070.

[36] M. A. Uddin, J. B. Joolee, A. Alam, Y. Lee, Human action recognition using adaptive local motion descriptor in Spark, IEEE Access 5 (2017) 21157–21167.

[37] L. Cheng, Y. Yu, X. Liu, J. Su, Y. Guan, Recognition of Human activities using fast and adaptive sparse representation based on wearable sensors, in: 16th IEEE Int. Conf. on Machine Learning and Applications (ICMLA), 2017, pp. 944–949. doi:10.1109/ICMLA.2017.00-32.

[38] Z. Sheng, C. Hailong, J. Chuan, Z. Shaojun, An adaptive time window method for Human activity recognition, in: IEEE 28th Canadian Conf. on Electrical and Computer Engineering (CCECE), 2015, pp. 1188–1192. doi:10.1109/CCECE.2015.7129445.

[39] H. K. Hameed, W. Z. W. Hasan, S. Shafie, S. A. Ahmad, H. Jaafar, An amplitude independent muscle activity detection algorithm based on adaptive zero crossing technique and

mean instantaneous frequency of the sEMG signal, in: IEEE Regional Symp. on Micro and Nanoelectronics (RSM), 2017, pp. 183–186. doi:10.1109/RSM.2017.8069133.