Pers Ubiquit Comput
DOI 10.1007/s00779-011-0455-4

ORIGINAL ARTICLE

Human motion recognition using a wireless sensor-based wearable system

John Paul Varkey · Dario Pompili · Theodore A. Walls

Received: 7 March 2011 / Accepted: 19 August 2011
© Springer-Verlag London Limited 2011

Abstract The future of human computer interaction systems lies in how intelligently these systems can take into account the user's context. Research on recognizing the daily activities of people has progressed steadily, but little focus has been devoted to recognizing jointly activities as well as movements in a specific activity. For many applications such as rehabilitation, sports medicine, geriatric care, and health/fitness monitoring the importance of combined recognition of activity and movements can drive health care outcomes. A novel algorithm is proposed that can be tuned to recognize on-the-fly range of activities and fine movements within a specific activity. Performance of the algorithm and a case study on obtaining optimal features from sensor and parameter values for the algorithm to detect fine motor movements are presented.

Keywords Body area networks · Activity recognition · Motion recognition · Classification · Support vector machines · Accelerometer · Gyroscope

J. P. Varkey · D. Pompili (✉)
Department of Electrical and Computer Engineering,
Rutgers University, New Brunswick, NJ, USA
e-mail: pompili@ece.rutgers.edu

J. P. Varkey
e-mail: johnpaul_varkey@cac.rutgers.edu

T. A. Walls
Department of Psychology,
University of Rhode Island, Kingston, RI, USA
e-mail: walls@uri.edu

1 Introduction

The future of human computer interaction systems lies in how intelligently these systems can take into account the user context, e.g., how well the data that it produces characterizes the user's current situation. Motion recognition is a key feature of many ubiquitous computing applications ranging from rehabilitation to health care. In general, motion recognition systems unobtrusively observe the behavior of people and the characteristics of their environments and, when necessary, take actions in response, ideally with little explicit user direction. Motion recognition aims at recognizing the actions of one or more users from a series of observations on the users' actions and the environmental conditions.

Sensor-based motion recognition integrates the emerging area of sensor networks with novel data mining and machine learning techniques to model a wide range of human motions. Human motion recognition systems composed of wirelessly connected sensor motes (equipped with accelerometers and gyroscopes) attached to different body sites enable a variety of applications such as rehabilitation, sports science/medicine, geriatric care, and health/fitness monitoring [2]. For example, such a system can be used to measure the effectiveness of active physiotherapy, to perfect techniques of sport persons, to remotely monitor and trigger emergency response for the elderly, and to help people lose weight by providing accurate estimates of their expended calories.

To understand human motion it is imperative to understand the difference between an *activity* and the *movements (or micro-activities)* that comprise it. A physical movement is a body posture/gesture that typically lasts for several milliseconds or seconds, while an activity lasts several minutes or hours and comprises of different physical

Published online: 11 September 2011

movements that may be repeated over time. For example, a "walking" activity would comprise of several short leg movements. There has been research [4, 12, 15, 20] on recognizing the daily activities of people such as whether a person is walking, jogging, standing, etc. However, most prior research has focused on activity recognition without directly considering the movements involved in that activity. Recognizing specific fine motor movements within activities of individuals will help provide a clear picture of the intensity of his/her activity. For example, in "walking" by knowing the number of steps (leg movements) taken by the person will help calculate the pace at which the person is walking. However, what makes movement recognition more challenging than activity recognition is that we are dealing with much shorter time scales.

To the best of our knowledge, there has not been a holistic approach proposed in the literature that addresses the challenges of using wireless wearable systems to recognize on the fly activities as well as the movements within a specific activity. We propose a simple, yet effective, window-based algorithm that can be tuned to recognize on-the-fly either activities or movements in a specific activity using a supervised learning approach based on Support Vector Machines (SVMs). Our approach involves a separate tuning phase to find optimal parameter values so as to maximize classification accuracy. In addition, we use multiple inertial data—linear acceleration values (collected using accelerometers) and angular rate of motion (collected using gyroscopes)—to recognize movements and activities. As using all the raw data (acceleration and angular rate) would be highly inefficient, meaningful features such as mean, standard deviation, maximum, peak-to-peak, root-mean-square, and correlation between values of accelerometer and gyroscope axes are extracted.

The remainder of this article is organized as follows. In Sect. 2, we review some of the existing work with respect our proposed work. In Sect. 3, we discuss the classification method for motion recognition using wearable sensors and then introduce the proposed window-based algorithm to recognize activities as well as movements. We further discuss the performance of the algorithm in recognizing activities and movements in Sect. 4. In addition, we present a case study on parameter tuning of the algorithm for movement recognition. Finally, in Sect. 5, we draw conclusions and discuss possible future work.

2 Related work

Previous work on activity recognition using acceleration values have considered features like mean [4, 12, 15, 20], standard deviation [12, 15, 20], maximum [4, 20], peak-to-peak [9], root-mean-square [12] of acceleration values and correlation of acceleration values between pair of axes of the accelerometer [4, 20]. However, in addition to the features collected from the accelerometer values we also extract and use features from angular rate values (gyroscope value) as knowing the orientation of various points of the body helps differentiate similar activities or movements. All of these works [4, 12, 15, 20] have focused on activity recognition without taking into account the movements involved. However, for some applications like behavioral study of patients and health monitoring, knowing the activities is just not quite enough. In such applications, knowing the movements involved would provide further information about the nature and intensity of the activity performed. Hence, in our work we focus on both activity and movement recognition, and we propose an approach that can be tuned to different timescales to be able to recognize both activities and movements and their starting and ending time instants.

Authors in [16] define a general framework for activity recognition by building upon and extending Relational Markov Networks. However, while the work is valuable, the authors introduced some constraints like one activity per location, which is improbable in real life. In [14], the authors introduce a sensor and annotation system for performing activity recognition in a house setting and used probabilistic models to learn the parameters of activities in order to detect them in future sensor readings. In [2], the authors discuss activity recognition results for stereotypical hand flapping and body rocking using data collected from children with autism in both laboratory and classroom settings; they also present a case study on the various challenges encountered when applying machine learning for recognizing activities. In [9], the authors show that movements have a grammatical framework such as a spoken language and introduce a linguistic framework for symbolic representation of inertial information by constructing primitives across the network of sensors for each movement and then using a decision tree. In [11], the authors formulate activity recognition as a pattern-based classification problem, and propose a novel emerging pattern-based approach to recognize sequential, interleaved, and concurrent activities. There is also some work on gesture recognition using accelerometers. In [18], the authors present uWave, a recognition algorithm using a single three-axis accelerometer to address the challenges in gestures recognition. In [28], the authors represent a hierarchical model to recognize both simple gestures and complex activities using a wireless body sensor network.

The computer vision community has conducted research on human motion recognition using time frames of a video sequence [8, 19]. However, the downside of these techniques is that processing video data is very costly and also an external infrastructure is required, e.g., (infrared) cameras, which may be biased by environmental conditions such as background light or heat. Also, such techniques cannot be directly applied to those scenarios that require privacy of the subject or of a third party (and this problem cannot simply be solved by blurring the images). Conversely, inertial sensors like accelerometers and gyroscopes are unbiased by environmental conditions and give a good accuracy for motion analysis. In addition, motion recognition systems using inertial sensors can be used in application where privacy is an important issue as these systems can be trained to recognize only specific predefined activities or movements.

Recently, activity recognition systems have found many application areas like health care and assisted living, industrial, entertainment, and gaming. In [3, 6, 13, 17, 25], the authors present systems for person fall detection as well as health threats by monitoring vital signs. A system has been proposed in [23] that uses information gathered from wearable and environmental sensors for tracking activities of workers in car manufacturing plants, e.g., to provide real-time feedback to the worker about upcoming assembly steps or to issue warnings when procedures are not properly followed. In [5], authors discuss a system that employs wearable inertial sensors combined with machine learning techniques to record, classify, and visualize the motion of dancers. However, none of these systems have employed a holistic approach to recognize on-the-fly activities and movements especially for behavioral study of people.

3 Proposed work

In order to recognize on-the-fly human physical activities such as walking, running, driving, dancing, gesturing or the specific movement types composing these activities, the proposed wearable system (1) collects raw data by distributed sensing and (2) employs supervised learning methods to identify activities and movements through localized computation. Interestingly, the system is *inherently unobtrusive* (not interfering with the user's day-to-day lifestyle) and ensures *privacy* unlike camera-based solutions (as it is trained to recognize only a predefined set of activities or movements). It also generates recording of user' behavior with little or no subject reactivity (user's behavior is unlikely to change as there is no or less interference by the system); these features enable research studies with higher internal consistency and support individually tailored treatment regimes especially when used for health monitoring [26, 27].

The proposed approach can be used for either activity recognition or movement recognition. According to our approach, we first pursue the activity recognition phase, which is then followed by the movement recognition phase that uses the same window-based algorithm (although it is separately trained and uses different parameters). The algorithm is based on machine learning, which requires training the system to the type of activities or movements that need to be recognized. First, we discuss the classification method used; then, we explain the various phases involved in our approach for human motion recognition. Finally, we describe and provide details of the window-based algorithm for recognizing activities or movements.

3.1 Classification method

Recognizing or classifying data is a common task in machine learning; if there are some data points each belonging to either one of two classes, the goal is to decide which class a new data point will be in. In SVMs [7], a data point is viewed as a p-dimensional vector; the objective is to separate such points using a (p-1)-dimensional hyperplane as it is shown in Fig. 1. A hyperplane can be defined as,

$$F(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} + b, \tag{1}$$

where $\mathbf{x}$ is the vector to be recognized and $\mathbf{w}$ is the normal vector to the hyperplane, which can be derived as,

$$\mathbf{w}^* = \sum_{i=1}^{l} \alpha^{i*} \cdot y^i \cdot \mathbf{x}^i,\ y^i \in \{-1, 1\}, \tag{2}$$

subject to the condition,

$$\alpha^{i*}[y^i \cdot (\mathbf{w}^{*T}\mathbf{x}^{\mathbf{i}} + b^*) - 1] = 0, \quad \forall \alpha^i \neq 0, \tag{3}$$

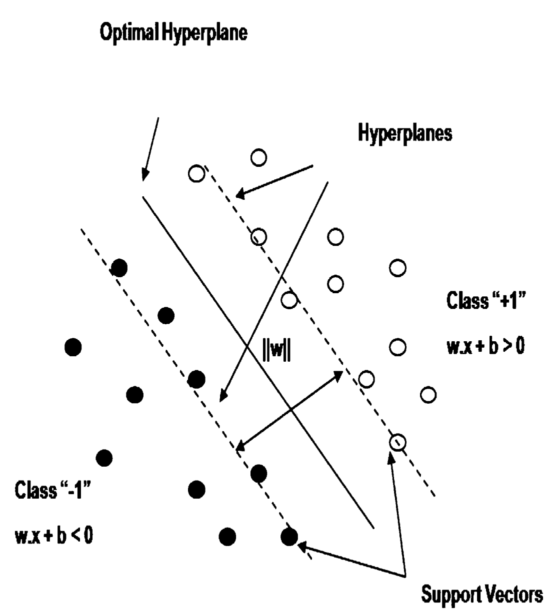

Fig. 1 Support vector machine (SVM)

where l is the number of support vectors (samples on the margin), α^i is the ith Lagrange multiplier, and b determines the offset of the hyperplane from the origin along the normal vector $\mathbf{w}$.

The function of the hyperplane in (1) is not suitable for solving more complicated, linearly non-separable problems and when dealing with more than two classes. Kernel function maps data into a high-dimensional space where the hyperplane can easily do the separation. We train the algorithm using the SVM-based machine learning toolbox available in Matlab (called "Spider" [22]) with RBF as the kernel (according to our experiments, this kernel is the most robust). To classify multiple classes, i.e., in our case multiple activities or movements, there are two common methods by which a classifier distinguishes—(1) "one-versus-all" (one of the labels to the rest) or (2) "one-versus-one" (between every pair of classes). We used "one-versus-one" method as it gave us the best results and also highlighted in [10] when the number of training samples is very large, the training can become problematic, and then "one-versus-one" strategy appears more suitable for practical use.

3.2 Problem formulation

Our approach for recognizing activities (or the movements in a specific activity) consists of three phases—*training*, *tuning*, and *motion recognition*, as shown in Fig. 2. In the training phase, the SVM is trained with sets of linear acceleration values (from accelerometers) and angular rate values (from gyroscopes) for each activity (or movement) types (depending on whether we want to recognize the activity or the movements in a particular activity). The tuning phase is then used to tune the parameters involved in our algorithm so as to improve the accuracy of recognition.

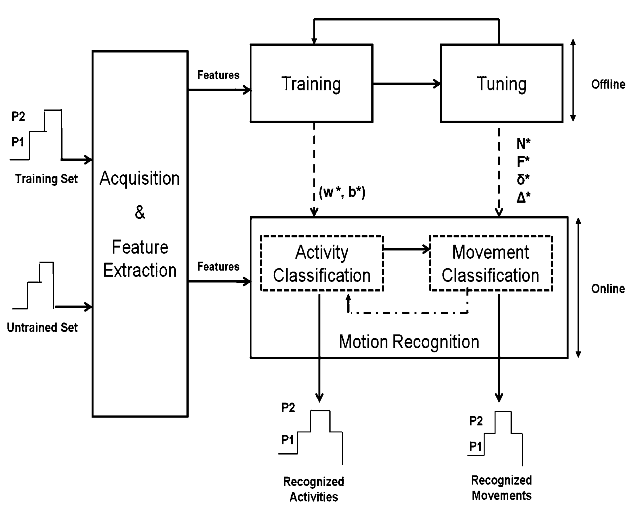

Fig. 2 Phases involved in motion recognition

We mainly tune the parameters when it involves recognizing the movements in a particular activity as movements have very shorter time scales compared to activities. In the motion recognition phase, once the SVM is trained with activity (or the movement) types, we use a novel window-based algorithm to recognize the activities or the movement types. As shown in Fig. 2, we have combined both activity and movement recognition under the phase *motion recognition*. In addition, in the motion recognition block we have shown a feedback from the movement recognition phase to the activity recognition phase, which can be exploited to improve the accuracy of the activity phase by using the outputs of the movement recognition phase.

Before explaining the proposed algorithm, let us define the parameters used to represent an activity or the movements within a specific activity in the training phase. We represent $\mathcal{D}$ as the training set, which is a set of activities or movements (depending on whether it is activity or movement recognition), which includes many observations of the same type of activities or movements. Hence, without loss of generality, $\mathcal{D}$ is a set of P observations of activities (or movements) irrespective of the types or classes such that each activity or movement $p = 1, 2, \ldots, P$. In addition, let ζ be the set of R types (or classes) of activities or movements considered for training, given as, $\zeta = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_r, \ldots, \mathcal{C}_R\}$, where $\forall p = 1, 2, \ldots, P$ and $p \rightarrow \mathcal{C}_r$ if the pth activity or movement is of type $\mathcal{C}_r$ (rth activity or movement type) considered for training, where $r = 1, 2, \ldots, R$. Each activity or movement p can then be associated with *any but only one* of the activity or movement type in ζ such that $R \ll P$.

Further, we represent $\mathcal{S}$ as the set of sensor nodes used to collect both linear acceleration and angular rate values from different body sites such that $s = 1, 2, \ldots, |\mathcal{S}|$. As mentioned before, we do not use the collected raw values as such for training but we extract a set of features from the raw values. We represent $\mathcal{F}$ as the set of features extracted from N sub-intervals (not from the entire time interval) from each axis of the accelerometer and gyroscope for each activity or movement p such that each feature $f = 1, 2, \ldots, |\mathcal{F}|$. Using all the parameters, v_p^s represent the pth activity or movement with respect to a sensor node s, and is given as, $v_p^s = [v_p^{sx}, v_p^{sy}, v_p^{sz}]$, where

$$v_p^{sx} = [f_1^{1x}, f_2^{1x}, \ldots, f_{|\mathcal{F}|}^{1x}; \ldots; f_1^{nx}, \ldots, f_{|\mathcal{F}|}^{nx}; f_1^{Nx}, \ldots, f_{|\mathcal{F}|}^{Nx}] \tag{4}$$

denotes the set of features extracted from the x-axis of the accelerometer and x-axis of the gyroscope attached to the sensor node s. Then, using the features extracted from the readings from all the sensor nodes pth activity or movement i.e., v_p can be represented as,

$$v_p = [v_p^1, v_p^2, \ldots, v_p^s, \ldots, v_p^{|\mathcal{S}|}]. \tag{5}$$

Finally, the entire training set $\mathcal{D}$ can be represented as,

$$\mathcal{D} = \{(v_1, \mathcal{C}_r^1), \ldots (v_p, \mathcal{C}_r^p), \ldots, (v_P, \mathcal{C}_r^P)\},\ \mathcal{C}_r^p \in \zeta. \tag{6}$$

Once the training phase is complete, we derive the normal vector $\mathbf{w}$ to the hyperplane as shown in (3) using the support vectors and the Lagrange multipliers [7] obtained from the SVM, which is then used to recognize the current activity or the movements in a particular activity during the recognition phase using the window-based algorithm. We define here the new concept of *confidence*, which forms the basis of our recognition algorithm. For a generic vector $\mathbf{x}$, we define its confidence as its distance from the hyperplane ($\mathbf{w}$, b) as,

$$d(\mathbf{w}, b; \mathbf{x}) = \frac{\mathbf{w} \cdot \mathbf{x} + b}{||\mathbf{w}||}. \tag{7}$$

This distance, $d(\mathbf{w}, b; \mathbf{x})$, indicates how "confident" the classifier is about the type or class of the vector $\mathbf{x}$: the larger d, the higher the confidence.

3.3 Proposed solution

We propose to use a window-based algorithm for recognizing the activities or movements. We first pursue the activity recognition phase, followed by movement recognition using the same window-based algorithm [24], but separately trained and optimized for activity and movement recognition. Figure 3 shows both the *logical* (a line representing the class) and *physical representation* (the raw signal, i.e., linear acceleration and angular rate values) of an activity and the movements in it. Hereafter, we will be using the logical representation for our further discussion as using the raw representation of activity and movements would be difficult and potentially misleading.

The algorithm uses two classification windows, a *main classification window* and a variable size *small classification window* that always moves only within the main classification window, as shown in Fig. 4. We define the intervals of activity (or movement) and main classification window to be as $[t_p^{\text{in}}, t_p^{\text{fin}}]$ and $[t_{\text{mw}}^{\text{in}}, t_{\text{mw}}^{\text{fin}}]$, respectively. In addition, we also consider two other intervals, $T^{\min}$ and $T^{\max}$, which we define as, respectively, minimum and maximum time intervals. Based on the demonstrated success for recognizing activities in previous works [4, 20] on accelerometer-based activity recognition using a window size of 32 samples with 16 samples overlapping between consecutive windows, we fixed the value of $T^{\max}$ and $T^{\min}$ to be $32 \cdot 0.05$, where 0.05 is the sampling time in seconds. The above definition holds only for activity recognition. While for movement recognition, $T^{\max}$ and $T^{\min}$ are, respectively, minimum and maximum time intervals among all the movement training samples irrespective of the movement types. Consequently, the following relationships hold,

$$T^{\min} = \begin{cases} \min_{1 \le p \le P} t_p^{\text{fin}} - t_p^{\text{in}} & \text{if } p \text{ is movement} \\ 32 \cdot 0.05 & \text{if } p \text{ is activity} \end{cases} \tag{8}$$

$$T^{\max} = \begin{cases} \max_{1 \le p \le P} t_p^{\text{fin}} - t_p^{\text{in}} & \text{if } p \text{ is movement} \\ 32 \cdot 0.05 & \text{if } p \text{ is activity} \end{cases} \tag{9}$$

We vary the size of small classification window interval T as,

$$T = T^{\min} + (h - 1) \cdot \delta, \tag{10}$$

where $h = 1, 2, \ldots, H$ and $H = \left\lfloor \frac{T^{\max} - T^{\min}}{\delta} \right\rfloor + 1$.

The main classification window interval (Fig. 4) is always set to $T^{\max}$ and is shifted over the entire time interval that needs to be recognized. For each h value, as given in (10), there is a small classification window interval T. The number of small classification windows within the main classification window H depends on $T^{\min}$ and the δ value. The small classification window is shifted

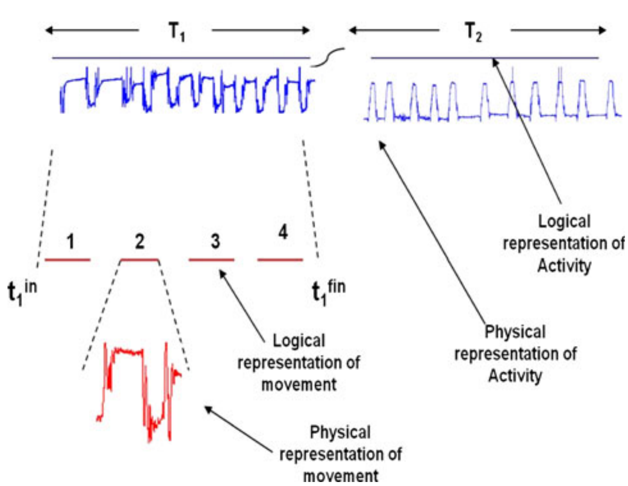

Fig. 3 Logical and physical representation of activities/movements

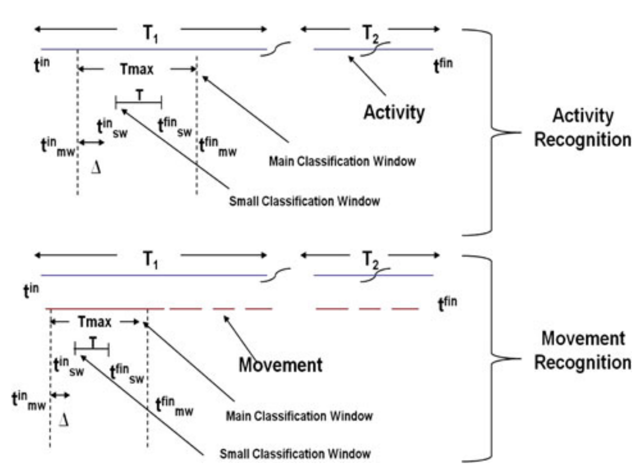

Fig. 4 Classification windows applied on activities and movements

by Δ within the main classification window for each T value, until it reaches the end of the main classification window. The number of such shifts is represented by K. For each shift within the main classification window, the confidence is calculated as in (7). Once the confidence for all the small classification windows within the main classification window is computed, we (1) select three intervals with the best confidence out of those initially considered, and (2) check for any (unfeasible) overlap in their time intervals. If there is any overlapping windows, we combine those time intervals and then compute the confidence for the combined interval. If there is no overlapping window, then we avoid those window intervals that have confidence less than the average confidence among all the small windows considered within the main classification window. The recognized activity/movement will be the class indicated by the SVM for the window interval that has best confidence. Once the activity/movement is recognized, the main classification window is shifted by the recently recognized time interval. Algorithm 1 summarizes the pseudocode of the described window-based algorithm.

Algorithm 1: Window-based Recognition Algorithm

```
H = ⌊(T^max − T^min)/δ⌋ + 1;
t^in_mw = t^in;
t^fin_mw = t^in_mw + T^max;
while t^fin_mw <= t^fin do
    for h = 1 to H do
        T = T^min + (h − 1) · δ;
        K = ⌊(T^max − T)/Δ⌋ + 1;
        for k = 1 to K do
            t^in = t^in_mw + (k − 1)Δ;
            t^fin = T + t^in_mw + (k − 1)Δ;
            extract features;
            find confidence;
    Take 3 recognized small windows with best confidence;
    if Overlapping intervals then
        find confidence of combined intervals;
    else
        Avoid intervals with confidence < average;
    Recognized Activity/ Movement = Class of window with
    best confidence;
    t^in_mw = t^fin;
    t^fin_mw = t^in_mw + T^max.
```

The performance of the algorithm depends on the following parameters—(1) Right feature set $\mathcal{F}$, (2) Number of sub-intervals N for feature extraction, (3) Number of small classification windows H that depends on δ, and (4) Number of shifts of small classification window K that depends on Δ. Hence, fine tuning is required before the algorithm can be used for recognition. The proposed window-based algorithm can be used to recognize activities or movements within a particular activity (once you know what the activity is). Because activities have much larger time interval compared to movements, we could simplify the algorithm for activity recognition by fixing the values of some of the parameters used. However, movement recognition involves shorter time scales and, therefore, optimal parameter values need to be identified using the tuning phase.

For tuning the system to the right values of $\mathcal{F}, N, \delta$, and Δ, we place the main classification window at the exact location of movements in the training set and try to recognize the movements by changing values of $\mathcal{F}, N, \delta$, and Δ. Hence, for tuning the system we feed the training data set itself and try to recognize the activities or movements with the training set. As a feedback metric to assess the optimality of these parameters, we use *cumulative misclassification ratio*, the ratio of total misclassification in time of recognized activity or movements with the total time interval of the actual activity or movement. Once the optimal values for $\mathcal{F}, N, \delta$, and Δ as indicated by starred values in Fig. 2 are known, we find the optimal support vectors to recognize movements.

For movement recognition, our algorithm works very well for activities in which movements are not overlapping and are time separable such as "Smoking," "Drinking," "Working out." Whereas for other activities like "Driving," as there are number of movements overlapping in time, the algorithm will still work fine but the system then needs to be trained for the many combinations of such overlapping movements.

4 Performance evaluation

The performance of the window-based algorithm in recognizing activities or movements in a specific activity as well as finding their starting and finishing instants is outlined below. As we mentioned, the algorithm can be used either for activity recognition or movement recognition, only one at a time or sequentially (first activity then movement recognition). To recognize activities, the SVM needs to be trained for the various types of activities. However, if the focus is on movement recognition within an activity, then the SVM needs to be trained for the various movement types within the concerned activity. First, we discuss how we collect acceleration and gyroscope values for various activities and movements. Then, we present the performance of the algorithm in recognizing activities and movements. Finally, we present a case study on the optimal parameter values in the window-based algorithm. To assess the performance of the algorithm for both activity and movement recognition, we used the *cumulative misclassification ratio* as index metric.

4.1 Data collection

For our study, we used Shimmer motes [21] for collecting linear acceleration (using accelerometers) and angular rate (using gyroscopes). A Shimmer mote has a triaxial accelerometer MMA7260Q made by Freescale and is capable of sensing accelerations ranging from $\pm1.5g$, $\pm2g$, and $\pm6g$, where $g = 9.81$ m/s^2. There is also a 3-axis gyroscope board having a full range of ±500°/s. The motes were placed on the right arm wrist and right foot of the subject as shown in Fig. 5. Each of these motes gathered linear acceleration (from accelerometer) and angular rate (from gyroscope) values at 20 Hz and transmitted it wirelessly to a destination node connected to a desktop that aggregated all the samples and extracted the features. For data collection we used two Shimmer motes; depending on the specific application, however, more sensors may be required for better accuracy.

4.2 Activity recognition

Activity recognition is the initial step in our approach. To show the performance of the window-based algorithm in recognizing various activities, we considered six different activities—"Walking," "Standing," "Writing," "Smoking," "Jacks," and "Jogging." An activity is "static" when there is no movement involved while performing that activity. A "dynamic" activity, on the other hand, requires movement in order to accomplish the activity. To validate our performance of the proposed algorithm we included three subjects (all male subjects), one with age 28 and others of age 26. We asked the three subjects to perform all the six activities in the order they preferred but at least 8 times. Although our experiments were carried out in a controlled environment in which subjects were asked to repeatedly perform specific movements (in the order they preferred) but we believe our method will be effective in less controlled environments when given a larger training set that includes a wider range of movements (i.e., in field trials).

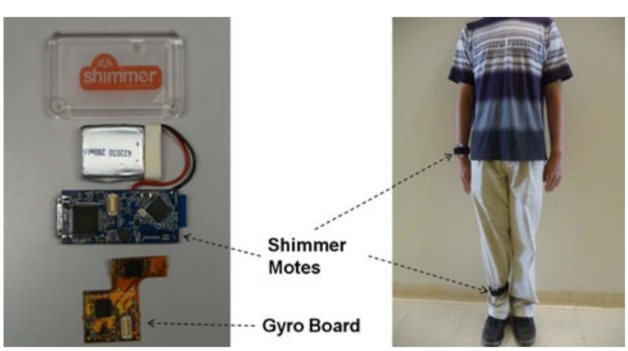

Fig. 5 Shimmer mote and gyro board; Shimmer motes attached to a subject

To evaluate the performance of activity recognition we did three case studies—(1) train separately for each subject and recognize activities of each subject—so as to discriminate against each subjects own potentially competing behaviors, (2) train for activities of all subjects as one and recognize the activities of each subject—as the approach used, tests for the potential "universality" that is, the device potentially could be trained on a large set of subjects and then work pretty well for everyone whose behavior is more or less the same way, and (3) train for one subject and recognize the activities of another subject—which is a extreme test of robustness where it really should not work at all for a precise activity, but remarkably it does. We show the test results of the above-mentioned three case studies in the form of a confusion matrix that shows the accuracy of recognition. In addition to the above-mentioned three case studies, to evaluate the performance of the algorithm in differentiating similar activities we also perform recognition on a new set of activities that are all very similar.

4.2.1 Case 1: Train SVM separately for each subject

In this case study, we trained the SVM separately for each subject one at a time and then tried to recognize the activities that each of the subject performed. Out of the 8 observations for each activity, 4 observations were used to train the SVM and the other 4 were used for testing the algorithm. We trained the SVM for say "Subject 1" using the first 4 observations and then tried to recognize the activities performed from the other 4 observations collected. Figure 6 shows the actual and recognized activities as well as their starting and ending time instants. "Subject 1" performed six activities in the order—"Smoking," "Walking," "Standing," "Writing," "Jogging," and "Jacks" as shown in Fig. 6. The cumulative misclassification ratio for the activities performed by Subject 1 is less than 1%. Similarly, Figs. 7 and 8 show the actual and recognized time instants of the activity performed by Subject 2 and 3, respectively, as well as the cumulative misclassification ratio. Subjects 2 and 3 performed the same six activities but in different order. The overall cumulative misclassification ratio is around 1.5%.

4.2.2 Case 2: Train SVM for all subjects at once

In this case study, we trained the SVM using the first 4 observations from all subjects all at once and then tried to recognize the activities performed by each subject using the next 4 observations. Figure 9 shows the actual and recognized activities as well as their starting and ending

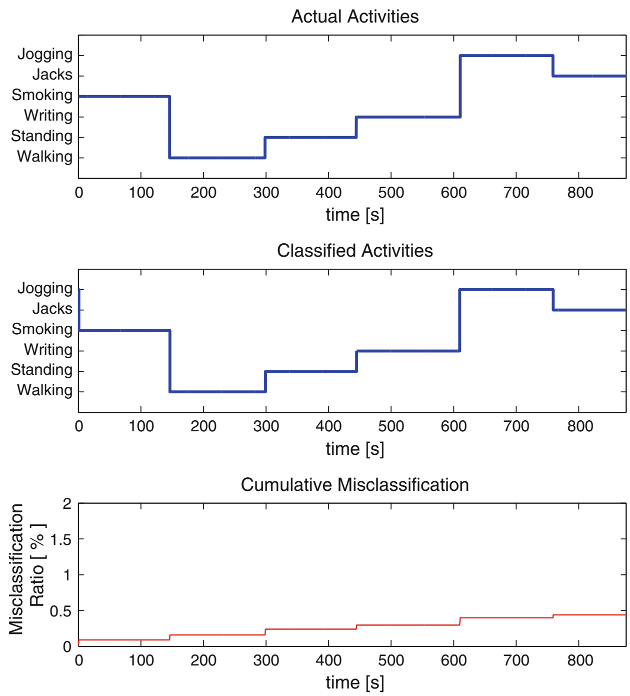

Fig. 6 Case 1: actual and recognized activities of Subject 1

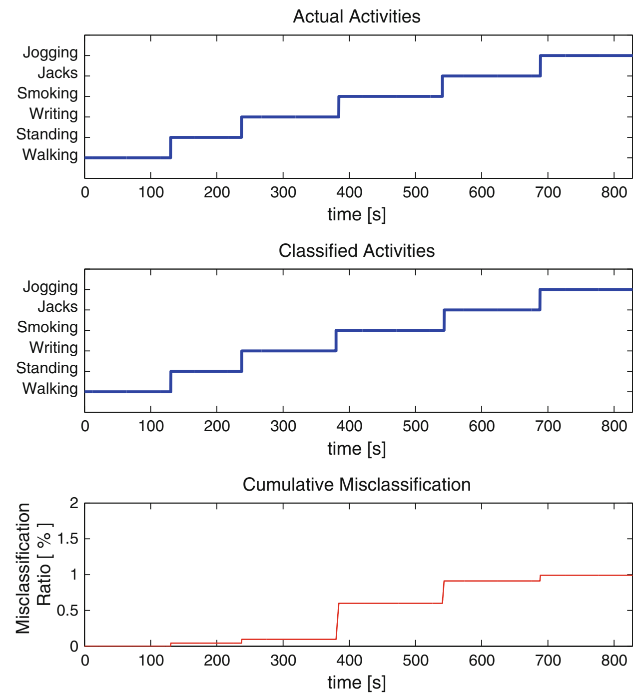

Fig. 8 Case 1: actual and recognized activities of Subject 3

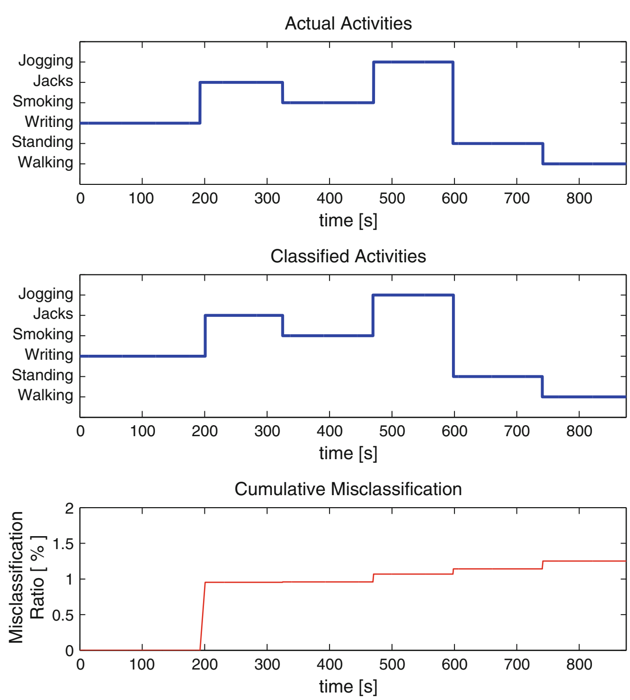

Fig. 7 Case 1: actual and recognized activities of Subject 2

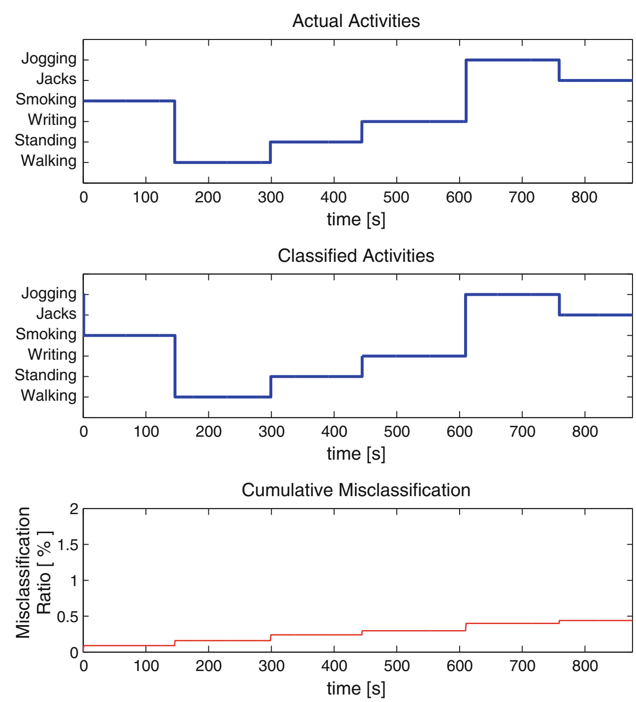

Fig. 9 Case 2: actual and recognized activities of Subject 1

time instants. The cumulative misclassification ratio over time is found to be less than 1%. Similarly, Figs. 10 and 11 show the actual and recognized time instants of the activities performed by Subject 2 and 3, respectively, along with cumulative misclassification ratio. The overall cumulative misclassification ratio is less than 1.5%.

4.2.3 *Case 3: Train SVM for one subject and recognize activities of another*

In this case study, we trained the SVM for one subject and then tried to recognize the activities performed by another

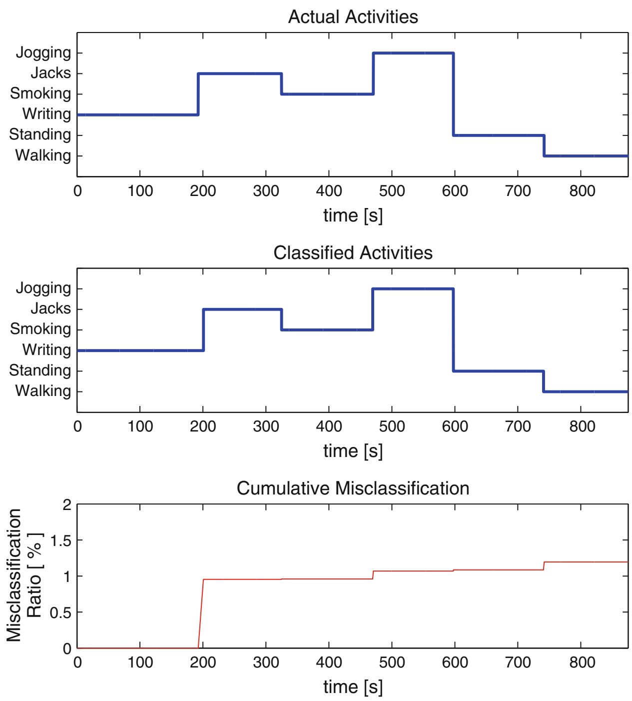

Fig. 10 Case 2: actual and recognized activities of Subject 2

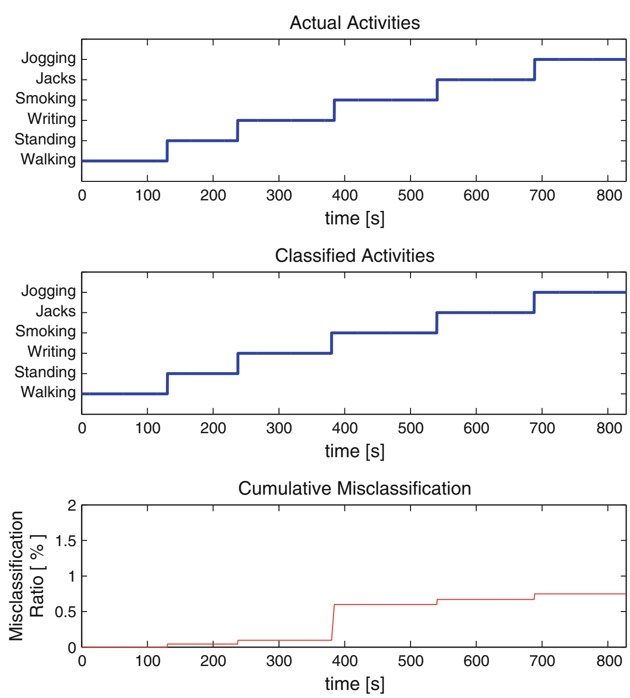

Fig. 11 Case 2: actual and recognized activities of Subject 3

subject. We trained the SVM using all the 8 observations collected from one subject and then tried to recognize the activities performed from all the 8 observations of another subject. Figure 12 shows the actual activities and their time instants as a result of the window-based algorithm run on

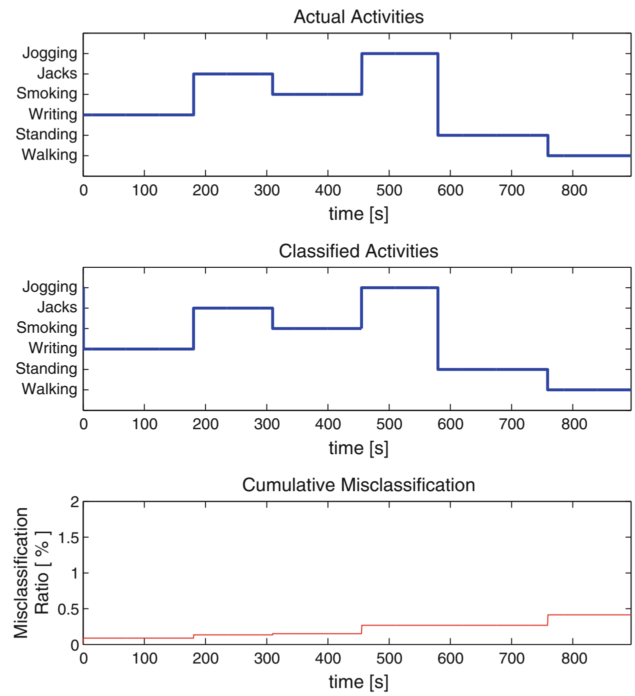

Fig. 12 Case 3: actual and recognized activities of Subject 2, SVM trained for Subject 3

"Subject 2" data when SVM is trained with data for "Subject 3." It is evident from the figure that the cumulative misclassification is around 1.6% (as expected, slightly higher than in the other two cases). Similarly, Fig. 13 shows the actual activities and recognized activities and their time instants, which "Subject 3" performed when the SVM is trained for "Subject 2." The overall cumulative misclassification is a bit higher than 1.6% which is more than for Case 1 and 2. The overall cumulative misclassification was also found to be around 1.6% for other scenarios where "Subject 1" data classified when trained by "Subject 2," "Subject 1" when trained by "Subject 3," "Subject 2" when trained by "Subject 1." Hence, we are showing in this paper the scenarios involving only "Subject 2" and "Subject 3."

Evaluation of the test results involved comparing the subjects actual activities with the recognized activities. If the recognized activity actually occurred during the appropriate time interval, then this outcome was recorded as a correct recognition; conversely, if a particular activity produced an unexpected recognition, then this outcome was considered an incorrect recognition. We show the test results for all the case studies done on the three subjects in a *"confusion matrix"* (see Table 1). It can be noticed that for Case 1, when SVM was trained separately for each subject one by one, all the activities were recognized accurately. However, for Case 2 and 3 there are some incorrect recognitions. In Case 2, when SVM was trained

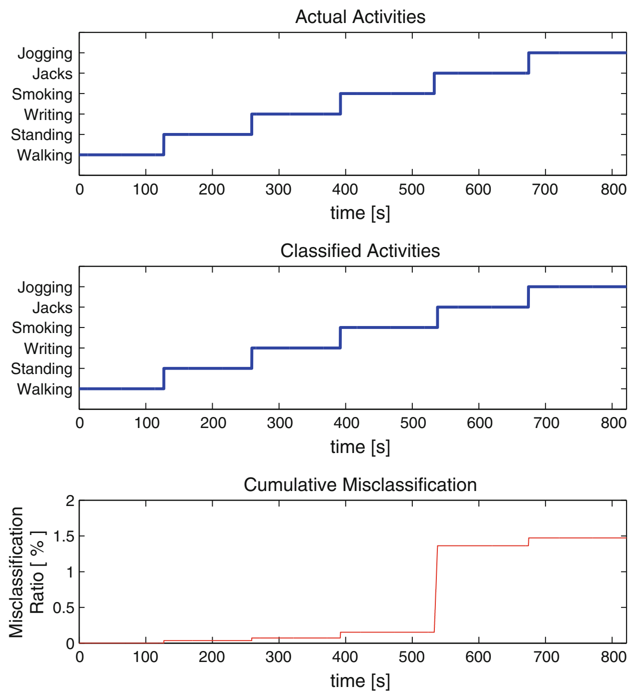

Fig. 13 Case 3: actual and recognized activities of Subject 3, SVM trained for Subject 2

for all the subjects, there was one misclassified result for activities—"Walking," "Smoking," and "Jogging." "Walking" was once incorrectly recognized as "Jogging" and also viceversa. This may have happened because of the change in pace while walking or jogging by the subjects. Similarly "Smoking" was incorrectly recognized as "Standing," maybe because there was a long pause between the puffs the subject took while smoking. In Case 3, when SVM is trained for one subject and the activities of another subject were used for recognition, there are some incorrect recognitions just like for Case 2. The overall accuracy of the system in recognizing the activities out of a total of 288 tests was found to be 97.2% and the lowest accuracy among all the case studies was found to be 91.6%. A presentation showing the video of the activities performed by one subject and the activities recognized by the system can be found in [1].

4.2.4 Classification of similar activities

In this case study, we classify a new set of activities that are all similar. The activities chosen were "brushing teeth," "writing," "eating with a fork," "typing," "smoking," and "drinking." These activities could all be described as the hand moving toward the face and back for a period of time. This was done to verify that the proposed algorithm could differentiate between activities that are similar but have subtle differences. We trained the SVM with two datasets from one subject and tested using three other datasets from the same subject. The subject was

Table 1 Confusion matrix

Cases	Actual	No. tests	Activity recognized						No. incorrect	Accuracy (%)
			Walking	Standing	Writing	Smoking	Jacks	Jogging		
Case 1	Walking	12	12	0	0	0	0	0	0	100
	Standing	12	0	12	0	0	0	0	0	100
	Writing	12	0	0	12	0	0	0	0	100
	Smoking	12	0	0	0	12	0	0	0	100
	Jacks	12	0	0	0	0	12	0	0	100
	Jogging	12	0	0	0	0	0	12	0	100
Case 2	Walking	12	11	0	0	0	0	1	1	91.6
	Standing	12	0	12	0	0	0	0	0	100
	Writing	12	0	0	12	0	0	0	0	100
	Smoking	12	0	1	0	11	0	0	1	91.6
	Jacks	12	0	0	0	0	12	0	0	100
	Jogging	12	1	0	0	0	0	11	1	91.6
Case 3	Walking	24	22	0	0	0	0	2	2	91.6
	Standing	24	0	24	0	0	0	0	0	100
	Writing	24	0	0	24	0	0	0	0	100
	Smoking	24	0	2	0	22	0	0	2	91.6
	Jacks	24	0	0	0	0	24	0	0	100
	Jogging	24	1	0	0	0	0	23	1	95.8
Total		288							8	97.2

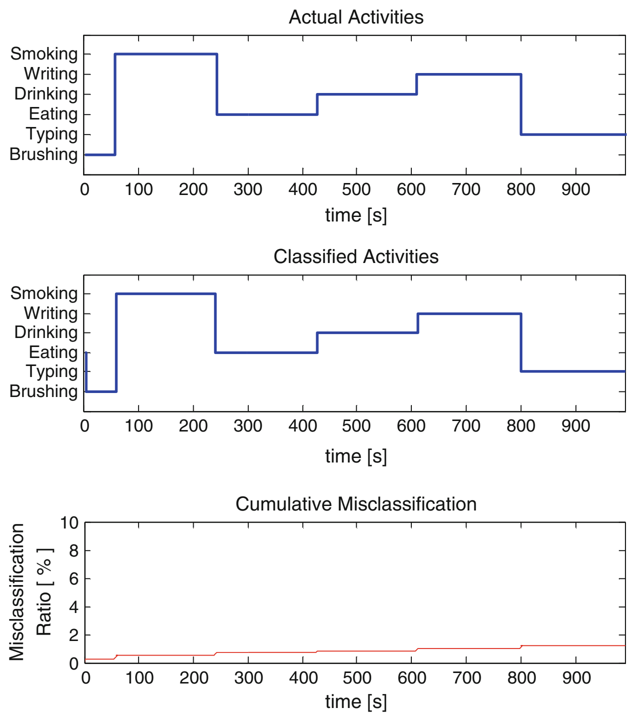

Fig. 14 Actual and recognized (similar) activities, performed by Subject 1 in realistic order, SVM trained for Subject 1

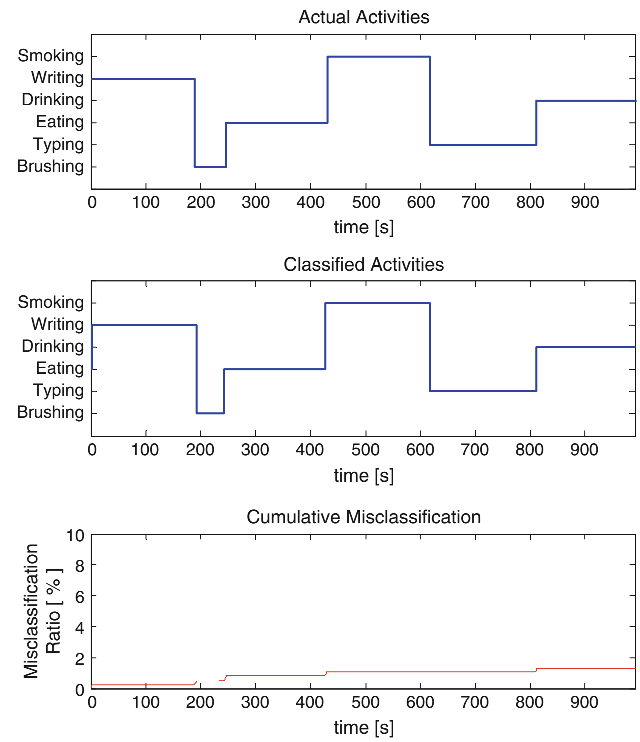

Fig. 15 Actual and recognized (similar) activities, performed by Subject 1 in random order, SVM trained for Subject 1

asked to perform the activities chosen in a realistic order, such as a morning routine. The subject smokes, then brushes his teeth. Then, the subject eats, then drinks, and finally writes and types. This series is shown in Fig. 14. In Fig. 15, the activities are ordered arbitrarily to verify that the algorithm is not affected by the order in which the activities are performed. In both cases we observe that the cumulative misclassification is extremely low (>2%).

4.3 Movement recognition

The second part of our approach involves recognizing movements in an activity. First, we present a case study on the best feature set for recognizing movements as well as other parameters such as the number of small classification windows considered and the number of shifts of a small classification window within the main classification window. Finally, using the best feature set and optimal values of the window-based algorithm, we discuss the performance of the algorithm in recognizing movements. To recognize each movement type within an activity, the SVM needs to be trained for each movement type. We consider that a movement in an activity is misclassified if either the movement type recognized is wrong or if the movement type recognized is correct but the recognized interval is less than 20% of the actual interval of the movement, which we call as *jitter*. As mentioned before, we used cumulative misclassification ratio to evaluate the performance of the algorithm. In addition, for movement recognition, which is much more complex than activity recognition due to the shorter time scales, we also used another index, *moving average misclassification ratio*, which we define as the ratio of misclassification of movements over a moving time window. For calculating this ratio, we considered a time window having an interval size of 10% of the activity time and shifted it over the activity.

For showing the performance of the algorithm, we take the case of recognizing movements within a "Smoking" activity as movements are not overlapping in time, which makes them separable. For proof-of-concept purpose, we assumed that a smoking activity comprises of a set of movements such as the "arm moving up" for smoking followed by the "arm moving down" after taking the puff, which are repeated until a cigarette gets over. To recognize the movements within one specific activity, we need to train the SVM separately for each movement type within that activity. Hence, we trained the SVM with the two movement types in the smoking activity. We took a set of 8 observations of smoking activity for training and another set of 8 for recognizing. We applied the window-based algorithm tuned for movement recognition in which we set around 3–4 min the expected time for a cigarette to be completely smoked by a subject. The subjects in the experiments are not real smokers but they are imitating the smoking movements and so the movements could be more similar than those of real smokers.

4.3.1 Best feature set

To identify the best feature set for recognizing movements within a specific activity, we considered in total six features. The best feature set for recognizing movements depends on the activity type. In $\mathcal{F}$, each feature is represented by a bit and features are considered in the order—mean, maximum, standard deviation, peak-peak, RMS, and then correlation. If a binary element in $\mathcal{F}$ is 0, it means that it is excluded, otherwise it is included. Hence, if $\mathcal{F} = [000110]$, then the features considered are peak-peak and RMS of linear acceleration and angular rate values. For six features we have a total of 63 combinations for $\mathcal{F}$. We took the values of N, δ, and Δ as 8, 3, and 2, respectively, for each combination of $\mathcal{F}$. In order to find the optimal $\mathcal{F}$, we placed the main classification window at position of occurrence of each of the movements in the training set and classified each of those movements. The optimal $\mathcal{F}$ would be the one that gives the least cumulative misclassification. Figure 16 shows the cumulative misclassification ratio for various $\mathcal{F}$ values from which we infer the optimal value of $\mathcal{F}$ is [000111]. Hence, the best feature set for recognizing movements in a smoking activity includes RMS, Peak-Peak, and Correlation of linear acceleration and angular rate values.

4.3.2 Optimal parameters

The performance of the classification algorithm depends on the N sub-intervals considered within a window to extract features. It also depends on the number of small classification windows H considered within the main classification window, which depends on δ, and on the number of shifts of small classification windows K within the main classification window, which depends on Δ. To find the optimal N, we considered 16 different values with $\delta = 3$, $\Delta = 2$, and $\mathcal{F} = [000111]$. Figure 17 shows that if N is too low, the cumulative misclassification is high; on the other hand, if N is too high, this would correspond to taking raw acceleration values of the movement, which produces too much data to process in quasi real time. In our experiments, it can be in observed that the optimal value for N is around 8. Similarly, it is also essential to find the optimal values of δ and Δ. Figure 18 shows cumulative misclassification ratio versus δ for various Δ. The optimal values of δ and Δ are 3 and 2, respectively, as the cumulative misclassification is the lowest for these values.

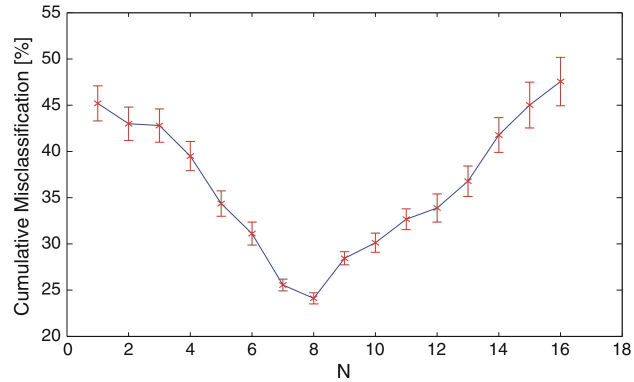

Fig. 17 Cumulative misclassification versus N @ $\delta = 3, \Delta = 2$

4.3.3 Movement recognition performance

Here, we show the performance of the algorithm in recognizing movements within a smoking activity. For recognizing the starting and finishing instants of the movements, we took for $\mathcal{F} = [000111]$, $N = 4, \delta = 3$, and $\Delta = 2$, which were identified as optimal values. Figure 19 show the actual and classified positions of both the movements involved in a smoking activity where "Movement 1" refers to "arm moving up" for taking puff while "Movement 2" refers to "arm moving down" after taking

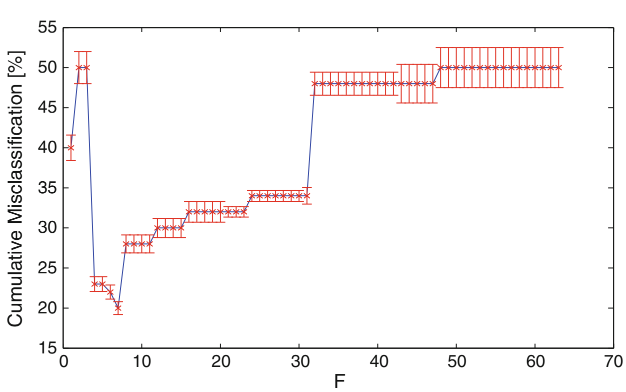

Fig. 16 Cumulative misclassification versus $\mathcal{F}$

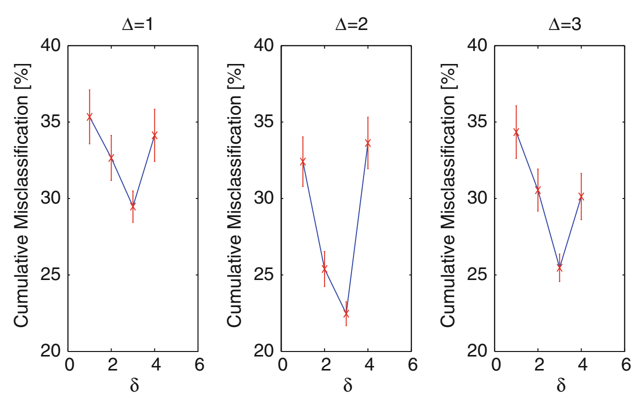

Fig. 18 Cumulative misclassification versus δ for variable Δ

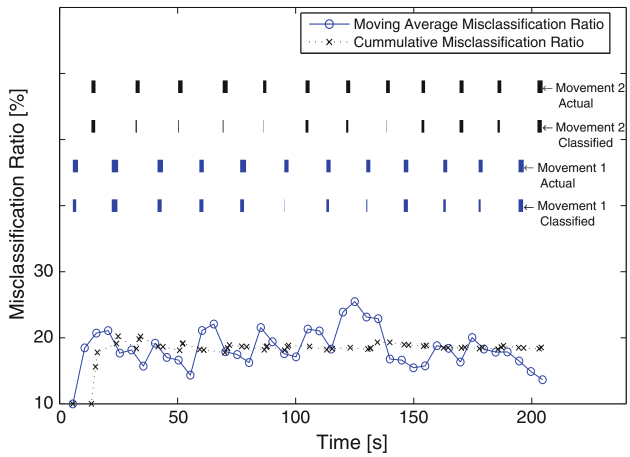

Fig. 19 Movement recognition in smoking activity 1

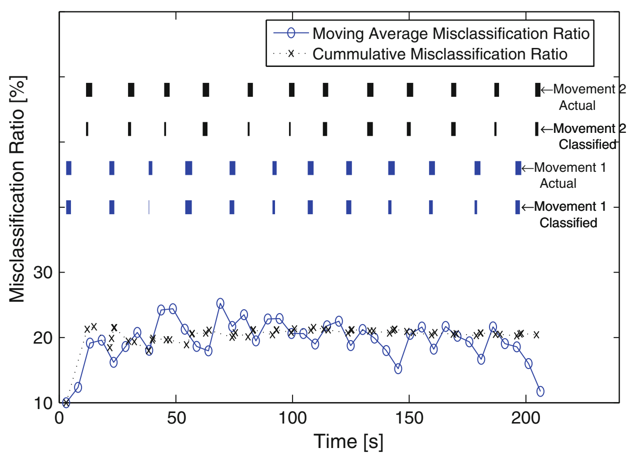

Fig. 20 Movement recognition in smoking activity 2

the puff. In Fig. 19, "Actual" refers to the actual position of the movements in the activity, whereas "Classified" refers to the time instants recognized by the algorithm. Within an activity, we represent a movement logically by a line of length equal to the time interval for which the movement occurs and different movement types are shown at different height within an activity. Figure 19 shows both the cumulative misclassification ratio and moving average misclassification ratio.

Figure 20 shows the recognized movements within another set of the smoking activity. Overall, the results show that cumulative misclassification is around 20%. Considering the fact that we did not use any a priori knowledge on the order of occurrence of the movements in the activity the misclassification rate is acceptable; furthermore, we considered the case of jitter in our misclassification calculation even if the movement is correctly classified. For example, in smoking activity, "Movement 2" usually follows "Movement 1" and so the order of occurrence of movements involved can be used to improve movement recognition (i.e., by filtering out unfeasible movement combination/order).

Misclassification for movement recognition, which is almost 20% is much greater than for the activity recognition, which is around 5–9% because of the following reasons: (1) in movement recognition we are dealing with shorter time scales and (2) multiple movements within an activity give way to multiple transitions, which are not negligible compared to the time scales of the individual movements. Last but not least, note that although we considered only six different activities and movements within "smoking activity," our approach can be generalized and applied for any number of activity or movement types separable in time (i.e., non overlapping).

5 Conclusion and future work

We proposed a novel learning-based algorithm that can be tuned to recognize on the fly either various activities or movements in a specific activity along with identifying their starting and finishing instants. We also identified the best set of features and optimal parameters values of the algorithm for improving the accuracy of fine motor movement recognition with a minimal sample of subjects and repetitions of the activities or movements. The results showed the accuracy of the algorithm to be around 91 and 80% for recognition of activity and movement, respectively.

As future work, we will further optimize the algorithm for movement recognition and train the system for various other daily-life activities. Further, we will use feedback from the movement recognition phase to the activity recognition phase to further improve the accuracy. In addition, we hope to move the approach into real-world trials for the classification of smoking and drinking behavior.

References

1. Activity Recognition and Vital Sign Monitoring, Cyber-Physical Systems (CPS) Lab, Rutgers University. http://nsfcac.rutgers.edu/CPS/demo.html
2. Albinali F, Goodwin MS, Intille SS (2009) Recognizing stereotypical motor movements in the laboratory and classroom: a case study with children on the autism spectrum. In: Proceeding of international conference on ubiquitous computing. Orlando, Florida
3. Anliker U, Ward J, Lukowicz P, Trster G, Dolveck F, Baer M, Keita F, Schenker E, Catarsi F, Coluccini L, Belardinelli A, Shklarski D, Alon M, Hirt E, Schmid R, Vuskovic M (2004) Amon: a wearable multiparameter medical monitoring and alert system. IEEE Trans Inf Technol Biomed 8(4):415–427
4. Bao L, Intille SS (2004) Activity recognition from user-annotated acceleration data. In: Proceeding of pervasive computing (PERVASIVE). Vienna, Austria

5. Barry M, Gutknecht J, Kulka I, Lukowicz P, Stricker T (2005) From motion to emotion: a wearable system for the multimedial enhancement of a butoh dance performance. J Mobile Multimedia 1(2):112–132
6. Bourke A, OBrien J, Lyons G (2007) Evaluation of a thresh-oldbased tri-axial accelerometer fall detection algorithm. Gait Posture 26(2):194–199
7. Christopher J, Burges C (1998) A tutorial on support vector machines for pattern recognition. Data Min Knowl Discov 2(2):121–167
8. Feng L, Yueting Z, Fei W, Yunhe P (2003) 3D motion retrieval with motion index tree. Comput Vis Image Underst 92(2): 265–284
9. Ghasemzadeh H, Barnes J, Guenterberg E, Jafari R (2008) A phonological expression for physical movement monitoring in body sensor networks. In: Proceeding of mobile Ad-hoc and Ssnsor systems (MASS). Atlanta, GA
10. Gidudu Anthony HG, Tshilidzi M (2007) Image Classification Using SVMs: one-against-one vs one-against-all. In: Proceedings of Asian conference on remote sensing (ACRS). Kuala Lumpur, Malaysia
11. Gu T, Wu Z, Tao X, Pung HK, Lu J (2009) epsicar: an emerging patterns based approach to sequential, interleaved and concurrent activity recognition. In: Proceedings of pervasive computing and communications (PerCom). Galveston, Texas
12. Heinz EA, Kunze KS, Junker SSH, Lukowicz P, Troster G (2003) Experimental evaluation of variations in primary features used for accelerometric context recognition. In: Proceedings of European symposium on ambient intelligence (EUSAI). Eindhoven, The Netherlands
13. Jafari R, Li W, Bajcsy R, Glaser S, Sastry S (2007) Physical activity monitoring for assisted living at home. In: Proceedings of international workshop on wearable and implantable body sensor networks (BSN). Germany
14. Kasteren TV, Noulas A, Englebienne G (2008) Accurate activity recognition in a home setting. In: Proceeding of international conference on ubiquitous computing. Seoul, South Korea
15. Kern N, Schiele B, Schmidt A (2003) Multi-sensor activity context detection for wearable computing. In: Proceedings of European symposium on ambient intelligence (EUSAI). Eindhoven, Netherlands
16. Liao L, Fox D, Kautz H (2005) Location-based activity recognition. In: Proceedings of neural information processing systems (NIPS). British Columbia, Canada
17. Liszka K, Mackin MA, Lichter MJ, York DW, Pillai D, Rosenbaum DS (2004) Keeping a beat on the heart. IEEE Pervasive Comput 3(4):42–49
18. Liu J, Wang Z, Zhong L, Wickramasuriya J, Vasudevan V (2009) uwave: accelerometer-based personalized gesture recognition and its applications. Pervasive Mob Comput 5(6):657–675
19. Muller M, Rder T, Clausen M (2005) Efficient content-based retrieval of motion capture data. ACM Trans Graph 24(3):677–685
20. Ravi N, Dandekar N, Mysore P, Littman ML (2005) Activity recognition from accelerometer data. In: Proceeding of American association for artificial intelligence (AAAI). Pittsburgh
21. Shimmer: http://shimmer-research.com
22. Spider: http://www.kyb.mpg.de/bs/people/spider
23. Stiefmeier T, Roggen D, Ogris G, Lukowicz P, Trster G (2008) Wearable activity tracking in car manufacturing. IEEE Pervasive Comput 7(2):42–50
24. Varkey JP, Pompili D (2009) Movement recognition using body area networks. In: Proceedings of IEEE global telecommunications conference (GLOBECOM). Honolulu, HW
25. Villalba E, Ottaviano M, Arredondo M, Martinez A, Guillen S (2006) Wearable monitoring system for heart failure assessment in a mobile environment. Comput Cardiol 237–240
26. Walls T (in press) Intensive longitudinal data. In: Little T, Nathan P (eds) Models for intensive longitudinal data. Oxford University Press, Oxford
27. Walls T, Schafer J (2006) Introduction. In: Walls T, Schafer J (eds) Models for intensive longitudinal data. Oxford University Press, New York, pp ix–xi
28. Wang L, Gu T, Chen H, Tao X, Lu J (2010) Real-time activity recognition in wireless body sensor networks: from simple gestures to complex activities. In: Proceedings of embedded and real-time computing systems and applications (RTCSA). Macau SAR, PRC