Pattern Classification with Meta-Cognition and Online Sequential Learning Algorithm

Skanda S. Bharadwaj
PES Centre for Int. Sys.
PES University Campus
Bangalore, India.
Skandabharadwaj94@gmail.com

R. Chandan Kumar
PES Centre for Int. Sys.
PES University Campus
Bangalore, India.
chandanrkumar619@gmail.com

B. N. Sumukha
PES Centre for Int. Sys.
PES University Campus
Bangalore, India.
sumukha.bn@gmail.com

Koshy George
PES Centre for Int. Sys.;
Dept. of Tel. Engg., PESIT
Bangalore, India.
kgeorge@pes.edu

***Abstract*—An integral part of modern day health-care is monitoring the physical activities of human beings. In this paper, we deal with automatic recognition of some daily activities based on signals measured using easily-available smart phones. We present a neural-network based methodology to classify these signals. In contrast to typical conventional techniques we use sequential processing of signals and circumvent pre-processing and feature extraction. In addition, we introduce meta-cognition to reduce the computations required during the training stage. We demonstrate that our approach yields satisfactory recognition accuracies.**

I. Introduction

Monitoring the physical activity of the elderly and special patients is becoming increasingly indispensable in health-care. These include patients with Parkinson's disease which result in motor disabilities, and patients with Alzheimer's disease which result in dementia and an inability for self-care. Typical activities such as sitting, standing and walking vary considerably from that of a healthy individual [1]. Accordingly, the importance of automatically recognising and monitoring the daily activities cannot be over emphasised.

Many techniques that have been developed in the past depend on the patient wearing multiple sensors. For example, in [2] and [3], a wireless body sensor network is used to gather information. The authors in [4] recognises the human activities from the data gathered with wearable inertial sensors. In a departure from these, sensors of mobile phones are used in [5] and [6] to obtain information about the movements. This appears to be less intrusive and puts patients at ease due to familiarity with the devices.

Methods for sensor-based activity recognition have been proposed by several researchers. Examples include the use of threshold-based classifiers in [8], binary decision trees in [7], Naïve Bayesian classifiers in [10] and hidden Markov models in [9]. Artificial neural networks were used in [11] and [12]. Semi-supervised and unsupervised learning are respectively considered in [13] and [14].

These techniques incorporate offline learning which involves complex data pre-processing and feature extraction. Accordingly, such techniques can be rather computationally expensive and time consuming. In contrast, in this paper, we consider online sequential learning using the data acquired from the sensors in a mobile phone to recognise some typical activities of human beings. The proposed methodology does not require pre-processing of signals and feature extraction. Since learning is sequential for this algorithm, it is better suited for real-time employments as signals are directly processed sample-by-sample. Specifically, we consider an ensemble of three three-layered feedforward neural networks (FFNNs) to classify the activities based on sequential processing of the raw data from the sensors. The networks are trained using the online sequential learning algorithm (OSLA). Moreover, meta-cognition is incorporated to monitor learning to reduce the training time.

The paper is organised as follows: The proposed methodology is presented in Section II. The algorithm to train the neural network is outlined in Section III followed by an introduction to meta-cognition in Section IV. Simulation experiments are described in Section V.

II. The Methodology

A. The Data

Smart phones are ubiquitous and have a number of built-in sensors. In this paper we consider the signals recorded from the accelerometer of a Samsung Galaxy SII mobile phone. These signals are made available as an open source by Smart Labs [6]. Recordings of six typical activities are available in this database. These are not recorded from patients but normal subjects who are made to walk, climb up and down a staircase, sit, stand and lie down. These activities are respectively denoted $\mathcal{C}_1$, $\mathcal{C}_2$, $\ldots$, $\mathcal{C}_6$. Evidently, the first three are dynamic and the last three are static.

The thirty subjects — all in the age group between 19 and 48 — were required to follow a protocol of activities while wearing the aforementioned smart phone. The details of the protocol are available in [6]. To summarise, they stood for 30 seconds, sat for 15 seconds and again stood for 15 seconds. After this, they lay down for 15 seconds, sat for 15 seconds and again lay down for 15 seconds. Finally, the subjects climbed up and down a staircase each of duration 15 seconds. Between each of these tasks there was a rest period of 5 seconds. All of these activities were carried out twice. In one set, the device was placed on the waist, and in the second set, the choice of the location was left to the subject. The signals are recorded for 192 seconds and the sampling frequency is 50 Hz. The

978-1-5090-6182-2/17/$31.00 ©2017 IEEE

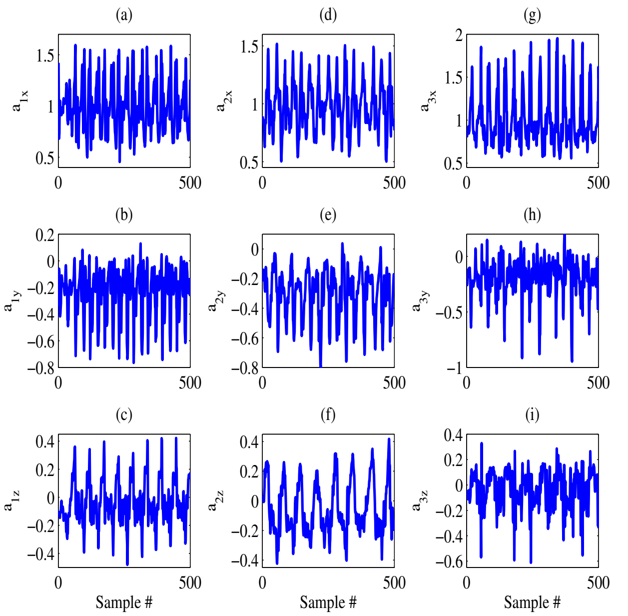

Fig. 1. Accelerations along the three axis for walking ($\mathcal{C}_1$: (a)–(c)), climbing up the staircase ($\mathcal{C}_2$: (d)–(f)), and climbing down the staircase ($\mathcal{C}_3$: (g)–(i)).

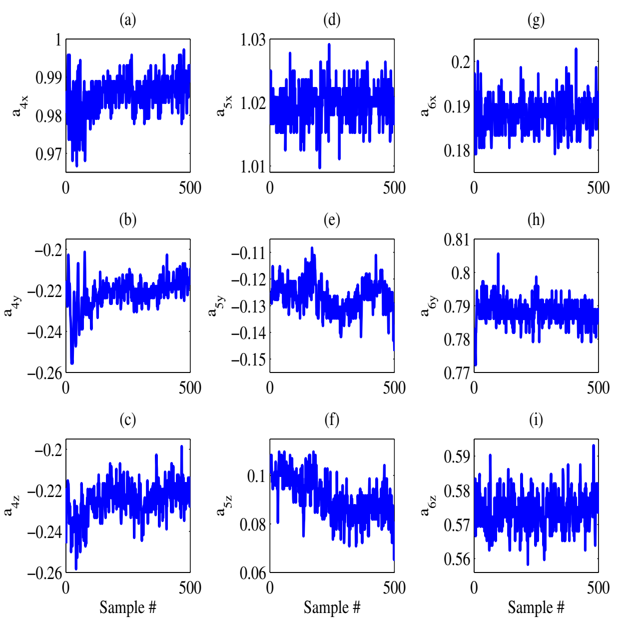

Fig. 2. Accelerations along the three axis for sitting ($\mathcal{C}_4$: (a)–(c)), standing ($\mathcal{C}_5$: (d)–(f)), and lying down ($\mathcal{C}_6$: (g)–(i)).

accelerations along the three axes of an inertial reference frame are measured for each subject by an accelerometer. From the database, we use a total of 856 signals. Typical acceleration profiles along the three axes are shown for each activity in Figures 1 and 2.

B. The Classification Process

The six classes mentioned earlier are to be differentiated and classified automatically in this paper. Further, this is to be carried out without any pre-processing and feature extraction. Towards this, we use feedforward neural networks (FFNNs) that are trained sequentially. In this manner the methodology proposed in this paper is more natural in that the human cognitive process works directly with the raw signals that fall on the sensors (e.g., visual, auditory and olfactory organs) without any explicit feature extraction.

We use an ensemble of three FFNNs — one for each axis — to classify the six human activities, as shown in Fig. 3. We denote these as $\mathcal{N}_x$, $\mathcal{N}_y$ and $\mathcal{N}_z$. The inputs to these networks are the present and previous values of the raw signals, and we emphasise sequential processing. Thus, these networks are essentially recurrent neural networks; i.e., static neural networks together with a tapped delay line, represented by the block **TDL** in the figure. In this paper, the length of the TDL is chosen to be twenty. Moreover, we consider a sliding window — depicted as the block **SW** in the figure — which slides by five samples resulting in an overlap of fifteen samples. This reduces the number of computations to some extent.

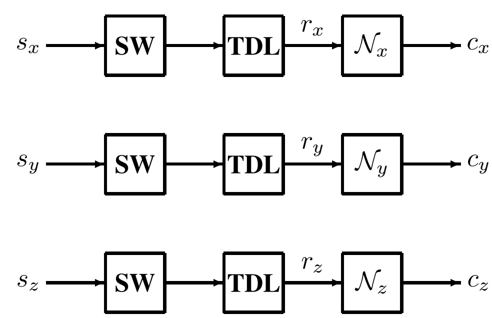

Fig. 3. An illustration of the proposed methodology. The blocks **SW** and **TDL** are respectively the sliding window and the tapped delay line. The artificial neural networks for the x, y and z axes are respectively the blocks $\mathcal{N}_x$, $\mathcal{N}_y$ and $\mathcal{N}_z$.

The class information from this ensemble is determined as follows: The three outputs c_x and c_y and c_z, respectively from $\mathcal{N}_x$, $\mathcal{N}_y$ and $\mathcal{N}_z$, are vectors belonging to $\mathbb{R}^6$ as there are six classes involved. The input to each neural network is a window of twenty samples. The output vectors of each neural network are concatenated into a matrix. Thus, assuming a total of M windows, we obtain three matrices of dimensions $6 \times M$. These matrices are denoted O_x, O_y and O_z, respectively for the three axes:

$$
\begin{aligned}
O_x &= \begin{pmatrix} c_{x,1} & c_{x,2} & \cdots & c_{x,M} \end{pmatrix} \\
O_y &= \begin{pmatrix} c_{y,1} & c_{y,2} & \cdots & c_{y,M} \end{pmatrix} \\
O_z &= \begin{pmatrix} c_{z,1} & c_{z,2} & \cdots & c_{z,M} \end{pmatrix}
\end{aligned}
$$

The averages along the six rows are then determined:

$$\mu_a = \frac{1}{M}\begin{pmatrix} \sum_{j=1}^{M} O_{a,1,j} \\ \sum_{j=1}^{M} O_{a,2,j} \\ \vdots \\ \sum_{j=1}^{M} O_{a,6,j} \end{pmatrix} = \frac{1}{M}\sum_{j=1}^{M} c_{a,j} \quad (1)$$

where a is x, y, or z depending on which of the three axes is being considered. Information about the class is present in the three vectors μ_x, μ_y and μ_z. In order to determine the class we consider a convex combination of the three networks:

$$\mu = \alpha_1\mu_x + \alpha_2\mu_y + \alpha_3\mu_z \quad (2)$$

where the constants are such that $0 \le \alpha_i \le 1$, $\alpha_1+\alpha_2+\alpha_3 = 1$, and $\mu = \begin{pmatrix} \mu_1 & \cdots & \mu_6 \end{pmatrix}^T$. Finally, the class is deduced as follows:

$$l = \arg\max_{1\le i\le 6} \mu_i \quad (3)$$

During the training phase, l is the target or desired class of the signal. The quantities $c_{d,x}$, $c_{d,y}$ and $c_{d,z}$ respectively denote the desired outputs of the three networks. Amongst the six elements of each vector, the position corresponding to the desired class label is assigned a unity value and the remaining elements assigned a value -1; thus, the elements of the desired output vector for a signal belonging to class $\mathcal{C}_i$ are given by

$$c_{d,a,m} = \begin{cases} 1, & \text{if } m = i \\ -1, & \text{if } m \ne i \end{cases} \quad (4)$$

In this paper we train the networks using the online sequential learning algorithm described in the next section.

III. Online Sequential Learning Algorithm

As shown in Fig. 3, we require three neural networks. Each one is a feedforward neural network with one hidden layer. The number of input nodes $m_0 = 20$ as the tapped delay line in Fig. 3 is of length 20, and the number of output neurons $m_2 = 6$ as the number of classes is six. If the number of neurons in the hidden layer is m_1, the network is denoted $\mathcal{N}_{a,m_0:m_1:m_2}$, where a is either x, y or z. The hidden layer is nonlinear with the activation function $\psi(v) = \tanh(v)$, and the output layer is linear. Let the synaptic weights connecting the input nodes to the hidden layer arranged as an array be denoted W_1, and the synaptic weights connecting the output of the hidden layer to the output layer arranged as an array be denoted W_2. From the equations provided later we can easily deduce the structures of W_1 and W_2.

Let the ordered pairs of data during the training phase be denoted $(r_{a,k}, c_{d,a})$, where $r_{a,k} \in \mathbb{R}^{m_0}$ and $c_{d,a} \in \mathbb{R}^{m_2}$, for $a = x$, y and z, and k indicates the sample number. Here, $r_{a,k}$ is the input to the network and $c_{d,a}$ is the desired class label. For each of the three axes, the computations in the forward pass may be summarised as follows:

$$\begin{aligned} \bar{y}_{1,k} &= \psi(W_1 y_{0,k}) && (5) \\ y_{2,k} &= W_2 y_{1,k} && (6) \end{aligned}$$

where

$$y_{0,k} \triangleq \begin{pmatrix} 1 \\ r_{a,k} \end{pmatrix}, \quad y_{1,k} \triangleq \begin{pmatrix} 1 \\ \bar{y}_k \end{pmatrix}.$$

In the online sequential learning algorithm (OSLA), the weight matrix W_1 is initialised randomly and the weight matrix W_2 is set to a zero matrix. Subsequently, the weight matrix W_2 is updated as follows: For $k \ge 0$,

$$\begin{aligned} P_{k+1} &= P_k - \frac{P_k y_{1,k+1} y_{1,k+1}^T P_k}{1 + y_{1,k+1}^T P_k y_{1,k+1}} && (7) \\ W_{2,k+1} &= W_{2,k} + e_{k+1} y_{1,k+1}^T P_{k+1} && (8) \end{aligned}$$

where the error in the a priori estimate

$$e_{k+1} \triangleq c_{d,a} - W_{2,k} y_{1,k+1}$$

and $P_0 = \frac{1}{\lambda} I_{m_1+1}$ with I_m an identity matrix of dimensions $m \times m$ and $\lambda > 0$. The weight matrix W_1 is not updated.

It is quite straightforward to show that the update equations (7) and (8) correspond to the recursive least squares solution to the Tikhonov-Phillips functional

$$J = \|W_2 Y_1 - Y_d\| + \lambda\|W_2\|, \quad (9)$$

where λ is also known as the regularisation parameter ([15]–[17]), and

$$\begin{aligned} Y_1 &= \begin{pmatrix} y_{1,1} & y_{1,2} & \cdots & y_{1,M} \end{pmatrix} \\ Y_d &= \begin{pmatrix} c_{d,a} & c_{d,a} & \cdots & c_{d,a} \end{pmatrix} \end{aligned}$$

The parameter λ provides a trade-off between minimisation of the training error and the synaptic weights of the output layer. Clearly, the minimiser of (9) is global.

Comments: (i) In contrast to the back propagation algorithm, the OSLA converges to the global minimum. Experience indicates that the convergence of OSLA is faster. It may be noted, however, that the back propagation algorithm is applicable to FFNNs with arbitrary number of hidden layers. (ii) The weight update equations (7) and (8) are similar to the weight update equations of the online sequential extreme learning machine (OSELM) proposed in [18], the sequential version of the extreme learning machine [19]. However, in contrast to OSELM, the initialisations are quite different. The different initialisation of OSLA results in much better performance compared to other sequential forms of learning including OSELM in the context of system identification and control [20], [21] and time-series prediction [22].

IV. Meta-cognition for Activity Classification

Learning in the natural world monitors and regulates itself. This is referred to as meta-cognition in the literature [23], [24], where it is defined as knowledge and control of cognition. Accordingly, it aids the learning process and the retention of knowledge. In this paper, we use meta-cognition to aid the process of classification of human activity following the spirit of [25]–[27] for meta-cognition. Specifically, meta-cognition

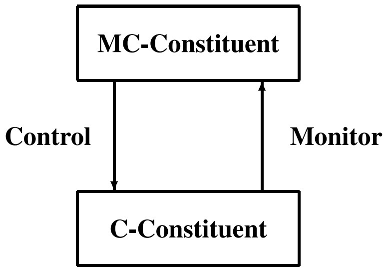

Fig. 4. The Nelson-Narens model for meta-cognition. The blocks **C-Constituent** and **MC-Constituent** respectively corresponds to the cognitive and meta-cognitive constituents.

is used to reduce the computations required during the training of the three networks $\mathcal{N}_x$, $\mathcal{N}_y$ and $\mathcal{N}_z$.

The Nelson-Narens model [28] shown in Fig. 4 is the basis on which meta-cognition is incorporated into the learning process. Evidently, it has two parts, the meta-cognitive and cognitive constituents represented as **MC-Constituent** and **C-Constituent**, respectively. The meta-cognitive constituent continually monitors the cognitive constituent. Based on some criteria, it affects the manner in which the cognitive constituent learns through the control signal.

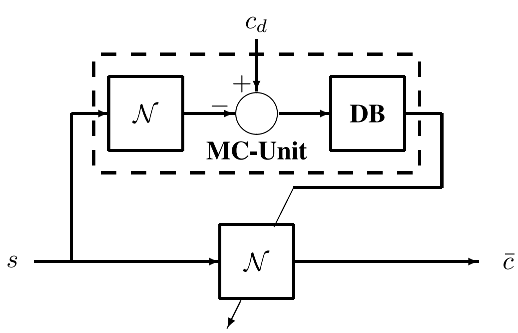

Fig. 5. The overall schematic for classification with meta-cognition. The blocks **DB** and $\mathcal{N}$ are respectively the decision block and the artificial neural network.

In our context, the manner in which the Nelson-Narens model is incorporated into the sequential classification process is shown in Fig. 5. The cognitive part consists of the ensemble of the three FFNNs denoted as $\mathcal{N}$ in the figure. Each of the networks has a hidden nonlinear layer and a linear output layer. The meta-cognitive part has a copy of the cognitive part. The signal to be classified is shown to the meta-cognitive constituent which determines the class to which it belongs. (It must be emphasised that the weights are not updated during the process; i.e., only the forward pass is considered.) The output of each network averaged over the M windows (i.e., μ_a) is mapped into the interval $[-1, 1]$ by scaling. Let this scaled output be denoted $\bar{c}_a$. Subsequently, the root-mean-square-error between $\bar{c}_a$ and the desired output $c_{d,a}$ is determined:

$$\varepsilon_a = \sqrt{\frac{1}{6}\sum_{i=1}^{6}(\bar{c}_{a,i} - c_{d,a})^2} \tag{10}$$

Clearly, $\varepsilon_a \in [0, 2]$. The meta-cognitive constituent monitors the learning process by tracking the value of ε_a, and decides upon one of the following actions for the network $\mathcal{N}_a$:

1) The synaptic weights of the network are updated.
2) The synaptic weights of the network are not updated.
3) The number of neurons in the hidden layer is increased.

In Fig. 5, these actions are taken by the decision block **DB** and results in the control signal in Fig. 4.

Suppose that $\epsilon_1 > 0$ and $\epsilon_2 > 0$ be two a priori chosen constants such that $\epsilon_1 < \epsilon_2$. For the network $\mathcal{N}_a$, if $\varepsilon_a < \epsilon_1$, then its synaptic weights are not updated. This avoids over-training and contributes to better generalisation property. In contrast, if $\epsilon_1 < \varepsilon_a < \epsilon_2$, then the same signal is presented to the cognitive constituent once again, and the weights of $\mathcal{N}_a$ are updated in accordance to (7) and (8). Accordingly, the novel information in the signal is learnt by the appropriate network. Finally, if $\varepsilon_a > \epsilon_2$, the complexity of the hidden layer of $\mathcal{N}_a$ is increased by increasing the number of neurons by one, and all the synaptic weights updated. This enhances its learning capacity. We note that the effect of a signal on the individual networks could be different. Clearly, the meta-cognitive part decides what is to be learnt and when.

The choice of the bounds ϵ_1 and ϵ_2 depend on the data. Determining this a priori is perhaps an open problem. If the bound ϵ_2 is a large value, meta-cognition has no major role. On the other hand, if set too low, the complexity of the network could increase arbitrarily. Thus ϵ_2 is a parameter that provides a trade-off between network complexity and performance. Similarly, a high value of ϵ_1 implies that more signals are not used to train the network, leading to a deterioration in the performance. In contrast, with a low value there is a possibility of over-training. Thus, ϵ_1 is a parameter that provides a trade-off between classification accuracy and generalisation performance. These bounds are arrived at experimentally.

We now present a technique to determine these bounds automatically and is adaptive in nature. When the first signal is shown we determine ε_a as explained earlier; we refer to this as $\varepsilon_{a,1}$. The bounds ϵ_1 and ϵ_2 are initialised as follows:

$$\epsilon_{1,a,1} = 1.1\,\varepsilon_{a,1}, \quad \epsilon_{2,a,1} = 0.9\,\varepsilon_{a,1} \tag{11}$$

Now, suppose that we are showing the pth signal from the training set. The bounds are then updated as follows:

$$\epsilon_{1,a,p} = 1.1 \min_{1\le i\le p} \varepsilon_{a,i}, \quad \epsilon_{2,a,p} = 0.9 \max_{1\le i\le p} \varepsilon_{a,i} \tag{12}$$

Comments: (i) Evidently, the bounds are chosen automatically, and adapted with every new signal in the training set. (ii) The coefficients 0.9 and 1.1 have been arrived at experimentally to maximise the performance. (iii) Note that initially $\epsilon_{1,a,1} > \epsilon_{2,a,1}$, violating the condition mentioned

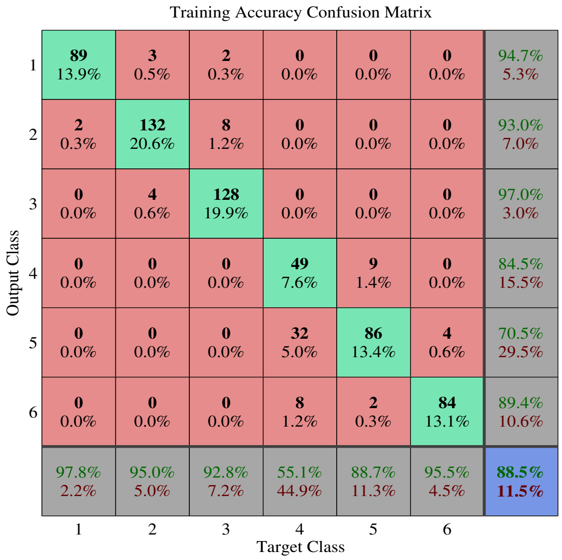

Fig. 6. The training phase confusion matrix for the 75:25 rule without meta-cognition.

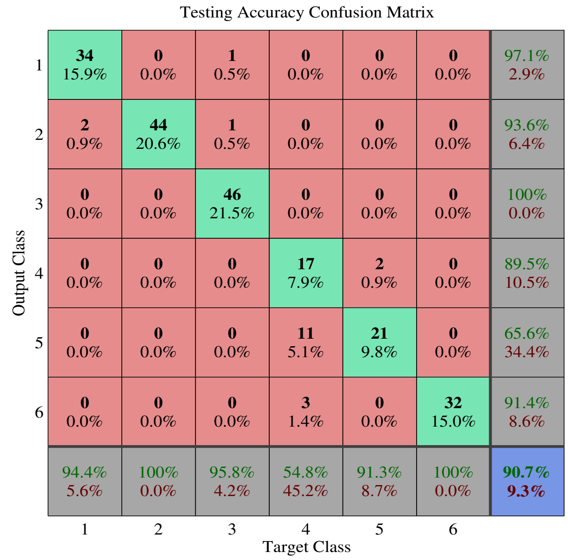

Fig. 7. The testing phase confusion matrix for the 75:25 rule without meta-cognition.

earlier. This violation allows exploration similar to that found in reinforcement learning algorithms. As observed later in Section V, within a few signals, the inequality is restored. (iv) Using meta-cognition in the context of pattern recognition is not new. In [25], the authors consider a radial basis function network that is trained sequentially. This idea is extended to a fully complex-valued radial basis function network in [26] and to an extreme learning machine classifier in [27]. However, features have to be extracted in order to train and test these networks. In contrast, we use a recurrent neural network that is trained and tested directly with the raw signals. Moreover, the authors in [25]–[27] do not consider bounds that are adapted.

V. Results and Discussions

The goal in this paper is to classify some of the activities of a human being based on the signals acquired through a mobile phone. We recall that the activities are dynamic (walking, climbing up and down the staircase) and static (standing, sitting and lying down). These are respectively referred to as classes $\mathcal{C}_1$ through $\mathcal{C}_6$. We use 856 signals from the database described in Section II. In this paper we consider both 4-fold cross-validation and 10-fold cross-validation. In the former case, we choose randomly 75% of the signals (i.e., 642) for training and the remaining 214 signals for testing. In what follows, this is referred to as the 75:25 rule. Similarly, for the 10-fold cross-validation or the 90:10 rule, 90% of the signals (i.e., 771) are randomly chosen for training, and the remaining 85 signals for testing.

The methodology consists of three FFNNs networks trained with OSLA. The number of neurons in the hidden layer for each network is experimentally found to be 120 so that the overall performance is maximised. Thus, the three networks are denoted $\mathcal{N}_{x,20:120:6}$, $\mathcal{N}_{y,20:120:6}$, and $\mathcal{N}_{z,20:120:6}$. For the

TABLE I
Average Classification Accuracies without Meta-cognition for the 75:25 Rule.

Class	Training Data	Testing Data
$\mathcal{C}_1$	91.15	91.14
$\mathcal{C}_2$	94.60	92.54
$\mathcal{C}_3$	97.27	97.54
$\mathcal{C}_4$	85.97	83.29
$\mathcal{C}_5$	72.16	71.47
$\mathcal{C}_6$	86.43	85.16
Overall	88.27	87.16

75:25 rule without meta-cognition the confusion matrices for training and testing are respectively shown in Figures 6 and 7. The classification accuracies averaged over 25 trials are depicted in Table I. The number of signals (and the percentage) that are correctly classified are the diagonal elements of the matrices. There are 642 signals used to train the networks. Of these, 89 (i.e., 13.9%) are correctly classified as $\mathcal{C}_1$, 132 (i.e., 20.6%) are correctly classified as $\mathcal{C}_2$, and so on. The classification accuracy is maximised by a careful choice of the constants α_1, α_2 and α_3 in (2). After several experiments, we chose $\alpha_1 = 0.4$, $\alpha_2 = 0.41$ and $\alpha_3 = 0.19$. Evidently, the accelerations along the x and y axes play a significant role. However, the acceleration along the z axis cannot be neglected. The off-diagonal elements in the confusion matrices represent incorrectly classified samples. The worst results are for $\mathcal{C}_5$. A number of subjects sitting down have been classified as standing with a few classified as lying down. A similar trend can be observed from Table II using the 90:10 rule. The 75:25 rule averaged over 25 runs resulted in average training

TABLE II
AVERAGE CLASSIFICATION ACCURACIES WITHOUT META-COGNITION FOR THE 90:10 RULE.

Class	Training Data	Testing Data
$\mathcal{C}_1$	90.10	90.95
$\mathcal{C}_2$	94.90	92.78
$\mathcal{C}_3$	97.31	97.32
$\mathcal{C}_4$	86.88	84.23
$\mathcal{C}_5$	72.11	71.10
$\mathcal{C}_6$	86.29	88.79
Overall	88.32	87.58

TABLE III
AVERAGE CLASSIFICATION ACCURACIES WITH META-COGNITION FOR THE 75:25 RULE.

Class	Training Data	Testing Data
$\mathcal{C}_1$	92.38	91.14
$\mathcal{C}_2$	94.76	93.04
$\mathcal{C}_3$	98.13	98.24
$\mathcal{C}_4$	82.13	82.05
$\mathcal{C}_5$	77.65	76.16
$\mathcal{C}_6$	93.43	92.16
Overall	89.23	88.12

and testing accuracies of 88.27% and 87.16%, respectively. The corresponding average accuracies with the 90:10 rule are 88.32% and 87.58%. From these metrics we conclude that the performances of the networks are reasonably consistent.

The achieved accuracy by the proposed method is comparable to those obtained in [6]. We note that an accuracy of 96% was achieved in [6] with a multi-class support vector machine after extracting 561 features after pre-processing the signals. This is in contrast to the methodology in this paper which does not require pre-processing or feature extraction. The poorer overall accuracy that is achieved in this paper relative to [6] is due to the class $\mathcal{C}_5$ which consists of subjects sitting down. This class was confused with those subjects who stood or lay down. Similar problems were observed in [6] and was attributed to the data acquisition process. When the subjects are dynamic (i.e., the subjects are walking or climbing), the classification accuracies achieved in this paper are 90.95%, 92.78% and 97.32% with the 90:10 rule. The authors in [6] achieve 96%, 98% and 99% respectively for classes $\mathcal{C}_1$, $\mathcal{C}_2$ and $\mathcal{C}_3$. These results show that the proposed method yields satisfactory classification accuracies.

The effect of meta-cognition for the 75:25 and 90:10 rules are given in Tables III and IV. In terms of overall classification accuracies there is a marginal improvement. A significant improvement is observed for $\mathcal{C}_5$ when meta-cognition is introduced. The 75:25 rule averaged over 25 runs resulted in average training and testing accuracies of 89.23% and 88.12%,

TABLE IV
AVERAGE CLASSIFICATION ACCURACIES WITH META-COGNITION FOR THE 90:10 RULE.

Class	Training Data	Testing Data
$\mathcal{C}_1$	91.21	92.10
$\mathcal{C}_2$	95.63	93.06
$\mathcal{C}_3$	97.70	98.64
$\mathcal{C}_4$	87.05	86.75
$\mathcal{C}_5$	72.93	74.65
$\mathcal{C}_6$	87.43	88.17
Overall	89.54	88.56

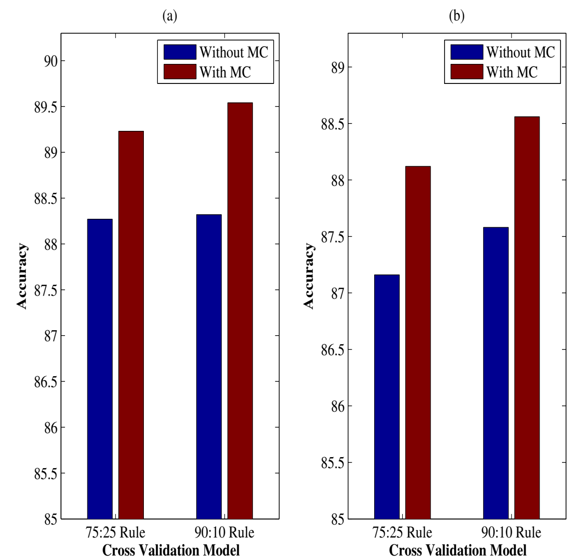

Fig. 8. Comparison of overall classification accuracies: (a) Training dataset. (b) Testing dataset. (The first and the third bars in each window correspond to the case when meta-cognition is not used.)

respectively. The corresponding average accuracies with the 90:10 rule are 89.54% and 88.56%. From these metrics we conclude that the performances of the networks are reasonably consistent. The averaged classification accuracies for different scenario with and without meta-cognition are compared in Fig. 8.

The introduction of meta-cognition reduces the training time considerably. The total numbers of discarded signals are 345 and 390 respectively for the 75:25 and 90:10 rules. That is, respectively 13.3% and 15.0% of the total number of training signals for the three networks are not used for training and with a marginal improvement in the classification accuracy. We recall that to compute the error only forward pass computations are considered and that the weights are not updated. Further, it was observed that for the 75:25 rule, the number of hidden neurons increased from 120 to 124, 128 and 127, respectively for the networks $\mathcal{N}_x$, $\mathcal{N}_y$ and $\mathcal{N}_z$. Similarly, for the 90:10 rule, the corresponding increases were 6, 10 and

TABLE V
TRAINING TIMES.

Learning	75:25 Rule	90:10 Rule
Without Meta-cognition	67.2940 sec.	92.8542 sec.
With Meta-cognition	63.5699 sec.	78.5162 sec.

9. The introduction of meta-cognition also serves the purpose of reducing the variance in the errors. The variance in the errors for the network $\mathcal{N}_x$ (averaged over 25 trials) without and with meta-cognition respectively are 0.00217 and 0.00149. The corresponding variances for $\mathcal{N}_y$ are 0.00257 and 0.00232, and for $\mathcal{N}_z$ are 0.00283 and 0.00269. This again shows that the networks are able to learn better with the introduction of meta-cognition. For all of these experiments the values of ϵ_1 for each axis is 0.8 and the values of ϵ_2 were 1.3, 1.41 and 1.43 respectively for the x-, y-, and z-axis.

The changes in these bounds when they are adapted in accordance with the heuristic adaptive rules presented in Section IV are shown in Fig. 9. Here, the errors ε_a computed for each training signal are shown as solid lines for the three axes. The bounds for each axis $\epsilon_{1,a}$ and $\epsilon_{2,a}$ are also indicated in these figures. For the first few signals we observe that $\epsilon_{1,a} > \epsilon_{2,a}$ in accordance with the manner in which they are initialised forcing the overall process to explore for the proper values of these bounds. Subsequently, $\epsilon_{1,a} < \epsilon_{2,a}$.

When these bounds are adapted the numbers of signals discarded were 312 and 336. That is fewer signals were discarded and hence a marginal increase in the training time when compared to the scenario wherein the bounds are fixed a priori. The changes in the classification accuracies are marginal. For the 4-fold cross-validation rule, the training and testing average accuracies averaged over 25 trials respectively are 89.74% and 88.28%. The corresponding accuracies for the 10-fold cross-validation rule are 89.54% and 88.56%. The largest increase in the neurons is seven along the y-axis for the 4-fold cross-validation. (These are summarised in Fig. 8.) Thus, when the bounds are initialised and adapted there is no significant change the performance. The main advantage is that there is no requirement of choosing these bounds a priori.

Comments: (i) The training times are as shown in Table V. (All simulations are performed on a MacBook Pro with an Intel I5 processor and 4 GB RAM.) We recall that the numbers of signals for the 75:25 and 90:10 rules respectively are 642 and 771 signals. Obviously, the latter rule takes significantly more time than the former rule. More importantly, meta-cognition reduces the number of signals that are used for training, and hence considerably reduces the training time. (ii) Attempts were made to use back propagation algorithm to train the networks in a manner similar to the philosophy adopted here: without pre-processing of signals and feature extraction, and to train sequentially. Networks with one and two hidden layers were used. However, the classification accuracy was significantly poorer than the one achieved here.

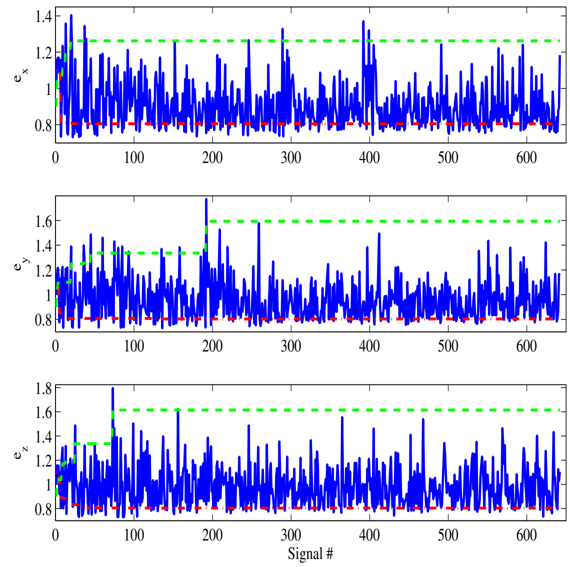

Fig. 9. The errors ε along x- and y- and z-axes as a function of the signal number shown as solid lines. The varying bounds ϵ_1 and ϵ_2 along each axis respectively shown as dash-dot and broken lines.

Accordingly, no statistical comparison of methods with the same underlying training philosophy was made.

VI. CONCLUSIONS

A methodology for recognising human activities was proposed in this paper. It requires no pre-processing of signals and feature extraction. An ensemble of three feedforward neural networks is used to classify the human activities. The classification accuracies achieved are comparable to other techniques that rely on feature extraction. The introduction of meta-cognition reduces the training time considerably with a slight improvement in the overall classification accuracy. The bounds required for meta-cognition can be made adaptive with only marginal changes in the performance. The proposed methodology is closer to the human cognitive process. The algorithm for supervised training is the online sequential learning algorithm. Work is in progress to statistically compare the proposed methodology with multiple datasets and other algorithms.

REFERENCES

[1] M. V. Albert, S. Toledo, M. Shapiro, and K. Kording, "Using mobile phones for activity recognition in Parkinson's patients," *Frontiers in Neurology*, vol. 3, pp. 1–7, November 2012.

[2] M. Bahrepour, N. Meratnia, Z. Taghikhaki, and J. M. Havinga, "Sensor fusion-based activity recognition for Parkinson patients," in *Sensor Fusion — Foundation and Applications*, C. Thomas, Ed. Rijeka, Croatia: InTech, 2011, pp. 171–190.

[3] S. Abbate, M. Avvenuti, P. Corsini, A. Vecchio, and J. Light, "Monitoring of human movements for fall detection and activities recognition in elderly care using wireless sensor network: A survey," in *Wireless Sensor Networks: Application-Centric Design*, G. V. Merrett and Y. K. Tan, Eds. Rijeka, Croatia: InTech, 2010.

[4] F. Attal, S. Mohammed, M. Dedabrishvili, F. Chamroukhi, L. Oukhellou, and Y. Amirat, "Physical human activity recognition using wearable sensors," *Sensors*, vol. 15, no. 12, pp. 31 314–31 338, December 2015.

[5] R. Jafari, W. Li, R. Bajcsy, S. Glaser, and S. Sastry, "Physical human activity monitoring for assisted living at home," in *4th International Workshop on Wearable and Implantable Body Sensor Networks (BSN 2007): RWTH Aachen University, Germany*, S. Leonhardt, T. Falck, and P. Mähönen, Eds. Berlin, Germany: Springer, March 2007, pp. 213–219.

[6] J. L. Reyes-Ortiz, A. Ghio, D. Anguita, X. Parra, J. Cabestany, and A. Catala, "A public domain dataset for human activity recognition using smartphones," in *Proceedings of the 21st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2013)*, Bruges, Belgium, April 2013, pp. 437–442.

[7] M. Ermes, J. Pärkkä, and L. Cluitmans, "Advancing from offline to online activity recognition with wearable sensors," in *Proceedings of the 30th Annual International IEEE EMBS Conference*, Vancouver, BC, Canada, August 2008, pp. 4451–4454.

[8] B. Coley, B. Najafi, A. Paraschiv-Ionescu, and K. Aminian, "Stair climbing detection during daily physical activity using a miniature gyroscope," *Gait and Posture*, vol. 22, no. 4, pp. 287–294, December 2005.

[9] E. M. Tapia, S. S. Intille, W. Haskell, K. Larson, J. Wright, A. King, and R. Friedman, "Real-time recognition of physical activities and their intensities using wireless accelerometers and a heart rate monitor," in *Proceedings of the 11th IEEE International Symposium on Wearable Computers (ISWC2007)*, Boston, MA, USA, October 2007, pp. 37–40.

[10] C. Zhu and W. Sheng, "Human daily activity recognition in robot-assisted living using multi-sensor fusion," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Kobe, Japan, May 2009, pp. 2154–2159.

[11] A. M. Khan, Y.-K. Lee, S. Y. Lee, and T.-S. Kim, "Human activity recognition via an accelerometer-enabled-smartphone using kernel discriminant analysis," in *Proceedings of the 5th International Conference on Future Information Technology*, Busan, S. Korea, May 2010.

[12] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "Human activity recognition on smartphones using a multiclass hardware-friendly support vector machine," in *Ambient Assisted Living and Home Care: 4th International Workshop, IWAAL 2012, Vitoria-Gasteiz, Spain, December 2012 Proceedings*, J. Bravo, R. Hervás, and M. Rodríguez, Eds. Berlin, Germany: Springer, 2012, pp. 216–223.

[13] D. Wyatt, M. Philipose, and T. Choudhury, "Unsupervised activity recognition using automatically mined common sense," in *AAAI05 Proceedings of the 20th National Conference on Artificial Intelligence*, Pittsburgh, PA, USA, July 2005, pp. 21–27.

[14] M. Stikic, K. Van Laerhoven, and B. Schiele, "Exploring semi-supervised and active learning for activity recognition," in *Proceedings of the IEEE International Symposium on Wearable Computers (ISWC 2008)*, Pittsburgh, PA, USA, September-October 2008, pp. 81–88.

[15] D. Phillips, "A technique for the numerical solution of certain integral equations of the first kind," *Journal of Association for Computing Machinery*, vol. 9, pp. 84–97, 1962.

[16] A. N. Tikhonov, "On solving incorrectly posed problems and method of regularization," *Doklady Akademii Nauk USSR*, vol. 151, pp. 501–504, 1963.

[17] A. N. Tikhonov and V. Y. Arsenin, *Solutions of Ill-Posed Problems*. Washington, DC, USA: V. H. Winston & Sons, 1977.

[18] N.-Y. Liang, G.-B. Huang, P. Saratchandran, and N. Sundararajan, "A fast and accurate online sequential learning algorithm for feedforward networks," *IEEE Transactions on Neural Networks*, vol. 17, pp. 1411–1423, 2006.

[19] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing*, vol. 70, pp. 489–501, 2006.

[20] K. Subramanian, S. G. Krishnappa, and K. George, "Performance comparison of learning algorithms for system identification and control," in *Proceedings of the 12th IEEE India International Conference (INDICON 2015)*, New Delhi, India, December 2015.

[21] K. George, K. Subramanian, and N. Seshadhri, "Improving transient performance in adaptive control of nonlinear systems," in *Proceedings of the 4th International Conference on Advances in Control and Optimization of Dynamical Systems (ACODS16)*, Tiruchirappali, India, February 2016, pp. 658–663.

[22] K. George and P. Mutalik, "Online time series prediction with meta-cognition," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN 2016)*, Vancouver, BC, Canada, July 2016.

[23] H. J. Flavell, "Metacognition and cognitive monitoring," *American Psychologist*, vol. 34, no. 10, pp. 906–911, 1979.

[24] J. H. Flavell, "Metacognitive aspects of problem solving," in *The Nature of Intelligence*, L. B. Resnick, Ed. Hillsdale, NJ, USA: Lawrence Erlbaum Associates, 1976, pp. 231–236.

[25] G. S. Babu and S. Suresh, "Meta-cognitive neural network for classification problems in a sequential learning framework," *Neurocomputing*, vol. 81, pp. 86–96, April 2012.

[26] R. Savitha, S. Suresh, and N. Sundararajan, "Meta-cognitive learning in a fully complex-valued radial basis function network," *Neural Computations*, vol. 24, no. 5, pp. 1297–1328, May 2012.

[27] R. Savitha, S. Suresh, and H. J. Kim, "Meta-cognitive learning algorithm for an extreme learning machine classifier," *Cognitive Computation*, vol. 6, no. 2, pp. 253–263, 2014.

[28] T. O. Nelson and L. Narens, "Metamemory: A theoretical framework and new findings," *Psychology of Learning and Motivation*, vol. 26, pp. 125–173, 1990.