Human Activity Recognition in Cognitive Environments Using Sequential ELM

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

Skanda S. Bharadwaj
PES Centre for Int. Sys.
PES University Campus
Bangalore, India.
Skandabharadwaj94@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*—Human activity recognition (HAR) and Extreme Learning Machines (ELM) are emerging fields of research. HAR investigates the behavioural attributes of humans and integrates that to an electronic system. An ELM is a fast learning algorithm, and overcomes the fundamental issue of slow training-error convergence that other algorithms such as the back propagation algorithm suffer. In this paper, we present the blend of the two fields by classifying the behavioural attributes of humans using Artificial Neural Networks (ANN) trained by Sequential Extreme Learning Algorithm (SELA). The algorithm is efficacious with a remarkable accuracy despite circumventing the vital job of pre-processing and feature extraction from signals that have been acquired from sensors.**

***Keywords*—Human activity recognition, extreme learning machines, sequential learning**

I. Introduction

Human Activity Recognition (HAR) has been of interest in the recent past and work is being carried that would automatically recognize these activities. The increase in the popularity of HAR is perhaps due to the ubiquitous presence of the so-called 'smart' devices which have helped develop cognitive and interactive environments. For example, in [1], several human behavioural attributes are taken into consideration to understand human behaviour. Integration of these attributes into devices such as smart phones helps in interaction with the environment.

At present, activity recognition is carried out with the information obtained from a camera [2], or by using sensors mounted on the body [3]. Vision-based techniques are based on methods such as optical flow models [4], Kalman filtering and Hidden Markov Models (HMM) [5]. Sensor-based activity recognition incorporates data mining, statistical modelling and machine learning techniques. We present an analysis using the latter approach.

In the context of activity recognition, there is very little work reported using semi-supervised and unsupervised learning (e.g., [6] and [7]). Supervised learning is predominant in this literature. For example, feedforward neural networks are used in [8] and multi-class support vector machine in [9]. Other techniques have also been considered. For example, predictive models like binary decision trees and threshold based classifiers respectively in [10] and [11], and probabilistic models like HMM and Naïve Bayes classifiers respectively in [12] and [13]. Some researchers have considered issues regarding implementations [14].

All the techniques mentioned above consider offline training of the acquired data. To the best knowledge of the authors, there appears to be no work reported that explores online sequential learning. From a practical viewpoint, online learning is better suited for real-time applications. Moreover, offline techniques rely on the features that are extracted from the signals in the dataset. Intuitively, this process is computationally expensive and time consuming.

The back propagation algorithm (BPA) is an algorithm that can be considered for sequential learning. However, it is well-known that BPA requires a rather long time to train the network. Further, it is quite likely that learning stops as it gets trapped in a local minimum. Huang and his co-workers proposed the Extreme Learning Machines (ELMs) that overcome these drawbacks [15]. Such machines are feedforward neural networks (FFNNs) with a single hidden layer. The synaptic weights corresponding to the input layer is made independent of the environment by keeping it fixed, and the synaptic weights of the output layer is determined as a solution to a least-squares problem. Evidently, the training time is much faster than BPA as the requirement for iterations is completely eliminated.

The authors in [16] proposed a sequential version of ELM and called it the online sequential extreme learning machine (OSELM). This algorithm requires the inversion of a matrix that is related to the data. However, it has been the experience in [17] that OSELM can perform rather poorly as this matrix is often singular or nearly singular. Another sequential variant called the sequential extreme learning algorithm (SELA) was considered in these papers, and have been observed to perform quite satisfactorily in the context of system identification and control.

The goal of this paper is two-fold: First, to propose a methodology that would provide comparable classification accuracy in the context of human activity recognition. The methodology does not require feature extraction and allows sequential learning. The second objective is to use

978-1-5090-1025-7/16/$31.00 ©2016 IEEE

the sequential variant SELA in the context of pattern classification. The methodology is explained in Section II and the learning algorithm in Section III. The results using an available dataset is presented in Section IV.

II. Methodology

The database of signals is chosen from an open source provided by Smart Labs [18]. This consists of six basic daily-life activities. Of these, three are static in posture and three are dynamic. The former has subjects standing, sitting or lying down, and the latter has subjects walking, climbing down a staircase or climbing up a staircase. Specifically, a group of 30 subjects in the age group 19–48 followed a protocol of activities whilst wearing a Samsung Galaxy SII smart-phone. Each subject began in the standing position and continued to do so for 30 seconds, followed by sitting during the next 15 seconds, and again standing for 15 seconds. The subjects then lay down for the next 15 seconds followed by intervals of 15 seconds each of sitting and then lying down. They then walked for 30 seconds, and then alternated (over three cycles) between climbing down the staircase and climbing up the staircase, each for 12 seconds. Between each task the subjects rested for 5 seconds. Each subject undergoes the protocol of activities twice; one with the device mounted on the waist, and for the other, the location of the sensor is left as a choice to the subject.

Therefore, the signals are measured for a total 192 seconds using the accelerometer and the gyroscope of the smart-phone. The sampling frequency is 50 Hz. Whilst the former provides the acceleration along the three axes of an inertial reference frame, the latter provides three angular velocities with respect to the same reference frame. Thus, there are six signals for each subject. A total of 856 signals were extracted from the dataset. A sample of the raw signals acquired from the sensors for the different activities are shown in Figures 1–3.

The objective in this paper is to differentiate and classify using artificial neural networks the six activities without any pre-processing and feature extraction. Moreover, the neural networks are trained sequentially. Thus, the proposed methodology resembles closer to the working of the human cognitive process wherein the signals (e.g., speech, visual, smell) are processed sequentially without any explicit feature extraction.

The schematic of the proposed methodology is given in Fig. 4. Three feedforward neural networks (FFNNs) are considered, one for each axis; these are denoted $\mathcal{N}_x$, $\mathcal{N}_y$ and $\mathcal{N}_z$. The outputs of the three networks are jointly considered for classification. Thus, we use a committee of FFNNs for classification of human activities. Moreover, we avoid entirely pre-processing the signals and extracting features from the signal. That is, the inputs to the networks are the raw signals, and the emphasis is on online sequential processing of data. To our best knowledge such an approach to pattern classification is novel.

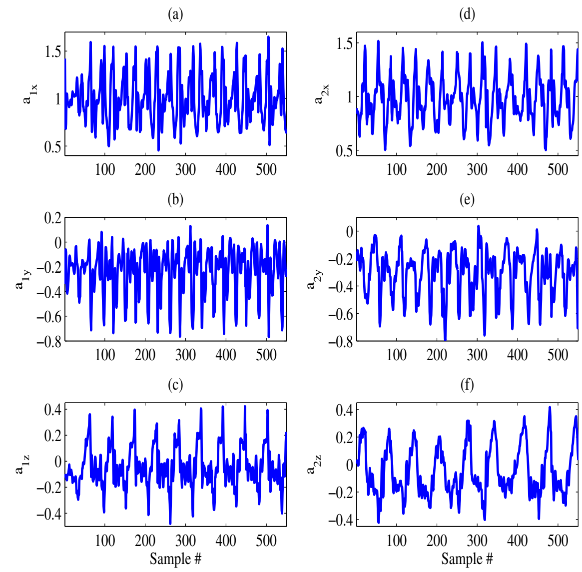

Fig. 1. Signals acquired from the sensors for different activities: (a)–(c) — Walking. (d)–(f) — Climbing up the stairs.

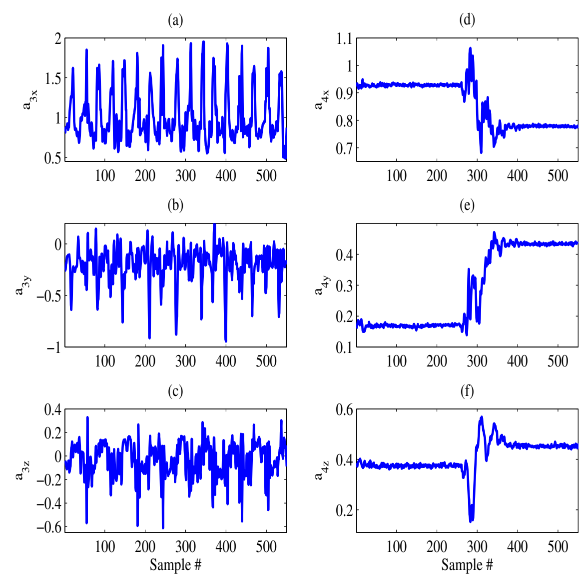

Fig. 2. Signals acquired from the sensors for different activities: (a)–(c) — Climbing down the stairs. (d)–(f) — Sitting.

The inputs to the FFNNs consist of the present and previous values of the signals. The block **TDL** in Fig. 4 represents the tapped delay line that achieves this. In this paper we consider a TDL of length 20. In order to reduce the computations, we slide the window of 20 samples by five, thereby creating an overlap of 15 samples. This is the block **SW** in Fig. 4.

Each network has six outputs corresponding to the six classes. Thus, for every window of 20 samples, each network yields an output vector that belongs to $\mathbb{R}^6$. Assuming a total of M windows, we concatenate the outputs

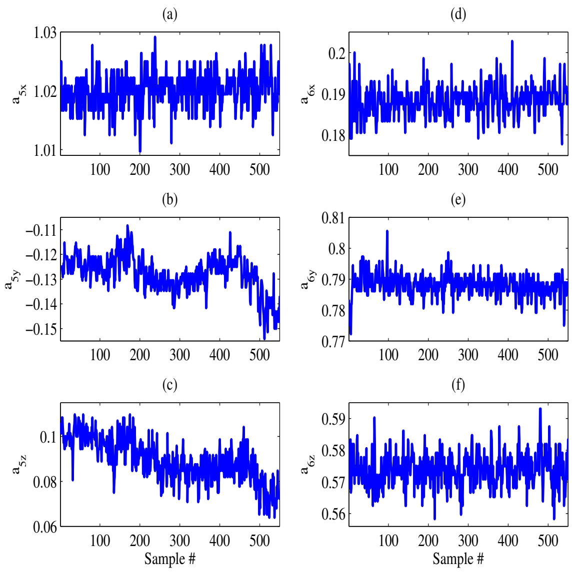

Fig. 3. Signals acquired from the sensors for different activities: (a)–(c) — Standing. (d)–(f) — Lying down.

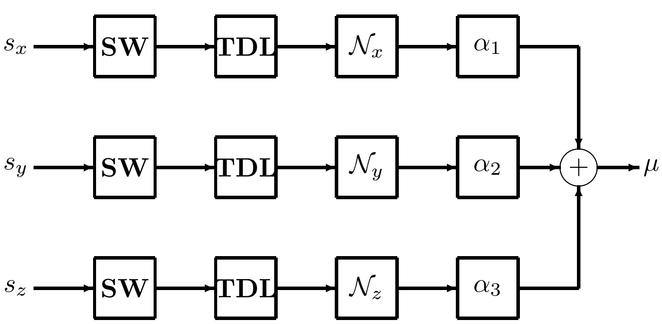

Fig. 4. The schematic for the proposed methodology. Here, **SW** is the sliding window, **TDL** is the tapped delay line, $\mathcal{N}_x$, $\mathcal{N}_y$ and $\mathcal{N}_z$ are the FFNNs for the x, y, and z axes, and α_1, α_2 and α_3 are constants.

to form a matrix of dimensions $6 \times M$. The output matrices for the x, y and z axes are respectively denoted O_x, O_y and O_z. The averages of the six rows are determined, and the class information extracted from these averages. Thus,

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

where a is either x, y, or z. Clearly, the location of the maximum of the six elements of μ_a is the class to which the signal belongs. In this paper we consider a convex combination of the outputs of the three networks to determine the class. That is,

$$\mu = \alpha_1 \mu_x + \alpha_2 \mu_y + \alpha_3 \mu_z$$

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

$$c = \arg \max_{1 \leq i \leq 6} \mu_i$$

During the training phase, c is the desired class of the signal.

In this paper, the networks are trained using a sequential variant of the extreme learning machine, referred to in the sequel as 'sequential extreme learning algorithm,' and denoted SELA. This is described in the next section.

III. Sequential Extreme Learning Algorithm

Consider an FFNN with a single hidden layer. The input layer has m_0 nodes, the hidden layer has m_1 neurons and the output layer has m_2 neurons. Such a network is denoted $\mathcal{N}_{m_0:m_1:m_2}$. Whilst the hidden layer has a nonlinear activation function, the output layer is linear. For an FFNN $\mathcal{N}_{m_0:m_1:m_2}$, the matrix of synaptic weights that connects the inputs nodes to the hidden layer is denoted W_1, and the matrix of synaptic weights that connects the hidden layer to the output layer is denoted W_2. Clearly, $W_1 \in \mathbb{R}^{m_1 \times (m_0+1)}$ and $W_2 \in \mathbb{R}^{m_2 \times (m_1+1)}$, where the bias is also taken into account. The activation function of the hidden layer is $\phi(v) = a \tanh(bv)$, and the output layer is a linear layer with unity gain. We denote the input to the network at instant k as $\mathbf{x}_k$ and the outputs of the hidden and output layer respectively denoted $\mathbf{y}_{1,k}$ and $\mathbf{y}_{2,k}$. The desired output at instant k is denoted $\mathbf{y}_{d,k}$. Clearly, $\mathbf{x}_k \in \mathbb{R}^{m_0}$, $\mathbf{y}_{1,k} \in \mathbb{R}^{m_1}$, and $\mathbf{y}_{2,k}$ and $\mathbf{y}_{d,k} \in \mathbb{R}^{m_2}$.

The extreme learning machine (ELM) is an algorithm to determine the synaptic weights using all the available data. Here, the weight matrix W_1 is randomly initialised. The outputs of each layer can be computed as follows:

$$\begin{aligned} \mathbf{y}_{1,k} &= \phi\left(W_{1,k} \bar{\mathbf{y}}_{0,k}\right) \\ \mathbf{y}_{2,k} &= W_{2,k} \bar{\mathbf{y}}_{1,k} \end{aligned} \tag{1}$$

where

$$\bar{\mathbf{y}}_{0,k} = \begin{pmatrix} 1 \\ \mathbf{x}_k \end{pmatrix}, \quad \bar{\mathbf{y}}_{1,k} = \begin{pmatrix} 1 \\ \mathbf{y}_{1,k} \end{pmatrix},$$

For the given training data is $\{\mathbf{x}_k, \mathbf{y}_{d,k}\}_{k=1}^{N}$, the outputs of the hidden layer for all the N inputs are computed using (1) and concatenated as follows:

$$Y_1 = \begin{pmatrix} \bar{\mathbf{y}}_{1,1} & \bar{\mathbf{y}}_{1,2} & \cdots & \bar{\mathbf{y}}_{1,N} \end{pmatrix} \in \mathbb{R}^{(m_1+1) \times N}$$

The output $\mathbf{y}_{2,k}$ of the FFNN to the N input patterns should match the desired value $\mathbf{y}_{d,k}$. That is,

$$Y_2 = W_2 Y_1 = Y_d, \tag{2}$$

where $Y_d = \begin{pmatrix} \mathbf{y}_{d,1} & \mathbf{y}_{d,2} & \cdots & \mathbf{y}_{d,N} \end{pmatrix} \in \mathbb{R}^{m_2 \times N}$. The only training required is to determine W_2 from (2):

$$W_2 = Y_d \, Y_1^T \left(Y_1 \, Y_1^T\right)^{-1}. \tag{3}$$

This is merely the least squares solution to (2) where the cost function $J_1 = \|W_2 Y_1 - Y_d\|$ is minimised.

The ELM assumes that all data is available a priori. When data arrives sequentially one-by-one, the weight matrix W_2 is updated as follows:

$$P_{k+1} = P_k - \frac{P_k \mathbf{y}_{1,k+1} \mathbf{y}_{1,k+1}^T P_k}{1 + \mathbf{y}_{1,k+1}^T P_k \mathbf{y}_{1,k+1}} \quad (4)$$

$$W_{2,k+1} = W_{2,k} + (\mathbf{y}_{m,k+1} - W_{2,k}\mathbf{y}_{1,k+1})\mathbf{y}_{1,k+1}^T P_{k+1} \quad (5)$$

with W_2 and P initialised as follows: $W_{2,0} = 0_{m_2 \times (m_1+1)}$ and $P_0 = \frac{1}{\lambda} I$.

Comments: (i) Using a derivation similar to that of the recursive least squares algorithm (RLS) it can be shown that $\lim_{k \longrightarrow \infty} W_{2,k+1} = W_{2,*}$, where $W_{2,*}$ is the minimiser to the functional $\|W_2 Y_1 - Y_m\| + \lambda \|W_2\|$. The parameter λ is also known as the regularisation parameter based on the minimisation of the Tikhonov-Phillips functional [19], [20]. (ii) The first sequential ELM algorithm was proposed in [16], referred to as the OSELM. This algorithm requires two phases. In the boosting phase, the initial weight matrix $W_{2,0}$ is determined using (3) with an initial chunk of data N_0. Subsequently, the weights W_2 is updated using (4) and (5). An assumption is that the inverse of $Y_{1,0} Y_{1,0}^T$ should exist; here, $Y_{1,0}$ is the output of the hidden layer with the initial chunk of data. This implies that this data should be linearly independent. In [17], this is discussed further, and it was demonstrated that this algorithm together with other sequential variants perform rather poorly in the context of system identification and control. In contrast to OSELM, our algorithm requires no boosting phase. Accordingly, it overcomes the drawbacks of OSELM, and it was shown to work well in applications related to identification and control of nonlinear systems.

IV. Results

As mentioned earlier, the objective is to classify some of the basic activities of a human. Specifically, we consider both static and dynamic situations. The former consists of subjects standing, sitting or lying down, and the latter consists of subjects walking, climbing up or down a staircase. The dataset consists of 856 signals. Of these, 796 signals are used to train the networks, $\mathcal{N}_{x,20:120:6}$, $\mathcal{N}_{y,20:120:6}$, and $\mathcal{N}_{z,20:120:6}$. (Thus each network has 20 input nodes, 120 neurons in the hidden layer, and 6 neurons in the output layer. This number of hidden neurons was chosen after many experiments.) The methodology was explained earlier in Section II and depicted in Fig. 4. The remaining 60 signals are used as test signals to determine the classification performance. The signals corresponding to walking, climbing up the stairs, climbing down the stairs, sitting, standing and lying down are respectively given the labels 1, 2, 3, 4, 5 and 6, and correspond to the classes $\mathcal{C}_1$, $\mathcal{C}_2$, …, $\mathcal{C}_6$.

The confusion matrices related to the training and testing dataset are shown in Figures 5 and 6. In these

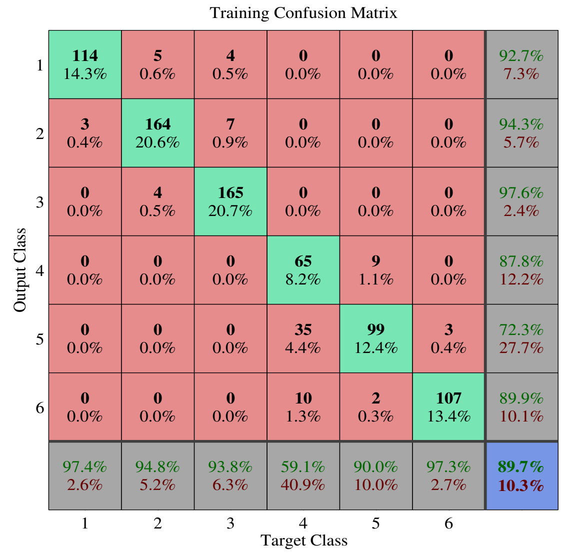

Fig. 5. The confusion matrix corresponding to the training phase.

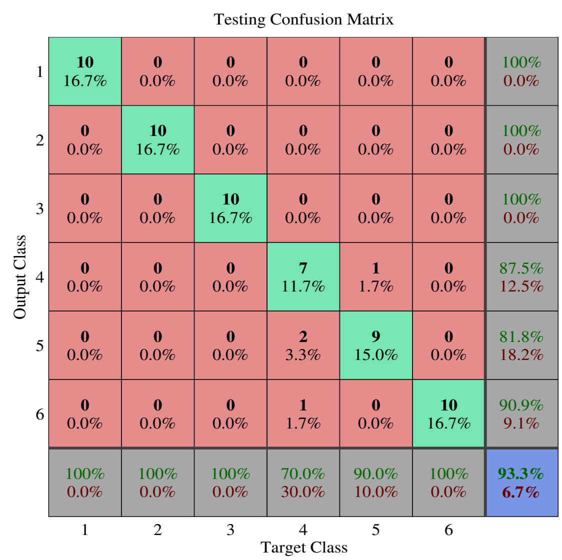

Fig. 6. The confusion matrix corresponding to the testing phase.

figures, the diagonal elements show the number and percentage of correct classifications. Thus, of the 796 signals used for training the networks, 114 of these have been correctly classified as $\mathcal{C}_1$ (i.e., 14.3%), 164 of these have been correctly classified as $\mathcal{C}_2$ (i.e., 20.6%), and so on. It may be noted that the percentage accuracy is maximised by carefully choosing the constants α_1, α_2 and α_3 in Fig. 4. After several experiments, we chose $\alpha_1 = 0.4$, $\alpha_2 = 0.41$ and $\alpha_3 = 0.19$. This implies that the signals along the x and y axes play a dominant role, and that the signals along the z axis cannot be ignored.

The off-diagonal elements represent incorrectly classified samples. Three samples that belong to $\mathcal{C}_1$ (i.e., 0.4% of the total samples) have been wrongly classified as belonging

TABLE I
Classification Accuracies for Training and Testing Data with 90:10 Rule.

Class	Training Data	Testing Data
$\mathcal{C}_1$	97.4	100.0
$\mathcal{C}_2$	94.8	100.0
$\mathcal{C}_3$	93.8	100.0
$\mathcal{C}_4$	59.1	70.0
$\mathcal{C}_5$	90.0	90.0
$\mathcal{C}_6$	97.3	100.0
Overall	89.7	93.3

to $\mathcal{C}_2$. Since no other samples of $\mathcal{C}_1$ are wrongly classified, the percentage of correct classification is 97.4% of 117 samples, and the percentage of incorrect classification is 2.6%. Similarly, there are a total of 173 samples in $\mathcal{C}_2$; of these, 164 are correctly classified (i.e., 94.8%) and nine are incorrectly classified (i.e., 5.2%). The latter includes five (i.e., 0.6% of the total samples) classified as $\mathcal{C}_1$ and four (i.e., 0.5% of the total samples) classified as $\mathcal{C}_2$. Each column therefore provides this information about a particular class. Accordingly, the last row gives the percentage of classification — correct and incorrect — for each class. For ease of reference, the percentage of correct classification is given in Table I. The worst results are for $\mathcal{C}_4$ where 35 have been incorrectly classified as $\mathcal{C}_5$ and ten have been incorrectly classified as $\mathcal{C}_6$. That is, a number of samples corresponding to subjects who are sitting have been incorrectly classified as subjects who are standing or lying down.

Finally, out of 123 classified as $\mathcal{C}_1$, 114 of these are correct (i.e., 92.7% of 123), and 9 are incorrect (i.e., 7.3% of 123). Similarly, out of 174 classified as $\mathcal{C}_2$, 164 (i.e., 94.3% of 174) are correctly classified and 10 are incorrectly classified (i.e., 5.7% of 174). The last column of the confusion matrix provides this information. Overall, 89.7% of all the samples have been correctly classified, and 10.3% of all the samples are incorrectly classified.

The confusion matrix related to the testing data is shown in Fig. 6. There are 60 samples in this dataset, and these are samples that the networks have not seen a priori. Clearly, the overall system has classified correctly 93.3% of all the samples, and classified incorrectly 6.7% of all the samples. That is only 4 of the samples are incorrectly classified.

The accuracy achieved by using three networks trained with SELA is comparable to that obtained using a multi-class support vector machine in [18], resulting an accuracy of 96%. However, in contrast to the methodology proposed in this paper, the authors in [18] extract first a total of 561 features after the signal has been pre-processed. The reason for our results to be poorer relative to that in [18] is due to $\mathcal{C}_5$, the class of subjects sitting down, and

TABLE II
Averaged Classification Accuracies for Training and Testing with 90:10 Rule.

Class	Training Data	Testing Data
$\mathcal{C}_1$	99.33	100.0
$\mathcal{C}_2$	90.83	91.71
$\mathcal{C}_3$	93.61	94.71
$\mathcal{C}_4$	62.14	61.92
$\mathcal{C}_5$	88.26	88.07
$\mathcal{C}_6$	95.99	96.17
Overall	89.11	88.28

TABLE III
Averaged Classification Accuracies for Training and Testing with 80:20 Rule.

Class	Training Data	Testing Data
$\mathcal{C}_1$	98.78	98.30
$\mathcal{C}_2$	90.78	89.87
$\mathcal{C}_3$	92.63	92.48
$\mathcal{C}_4$	61.32	59.06
$\mathcal{C}_5$	88.21	89.69
$\mathcal{C}_6$	96.28	94.73
Overall	88.77	87.63

the possibility of confusing this with subjects standing or lying down. It may be noted that similar issues were observed in [18]. (In that paper, the authors attributed this to the physical location of the smart-phone.) However, they had a greater problem with the dataset of subjects standing compared to that of subjects sitting down, which is different from that observed in this paper. For dynamic situations where the subjects are walking or climbing, the classification accuracy achieved in this paper is 100%. In contrast to this, the authors in [18] achieve 96%, 98% and 99% respectively for classes $\mathcal{C}_1$, $\mathcal{C}_2$ and $\mathcal{C}_3$.

Comments: The aforementioned results are for segregating the data into training and testing pairs using the 90:10 rule. For the same segregation, the results averaged over 25 trials are given in Table II. Other ways to segregate the data were also considered. The results averaged over 25 trials for 80:20 and 75:25 rules of segregation respectively are shown in Tables III and IV. Evidently, the averaged classification accuracies are fairly consistent. They do not appear to be affected by either the manner in which the data are segregated. Moreover, the effect of randomising the weight matrix W_1 appears to have only little effect on the classification results.

For our final comparison, we consider networks trained using the conventional back propagation algorithm using batch processing. We obtain a classification accuracy of 99.7%, which is better than even the multi-class sup-

TABLE IV
AVERAGED CLASSIFICATION ACCURACIES FOR TRAINING AND TESTING WITH 75:25 RULE.

Class	Training Data	Testing Data
$\mathcal{C}_1$	98.95	98.48
$\mathcal{C}_2$	90.69	88.85
$\mathcal{C}_3$	93.76	92.54
$\mathcal{C}_4$	62.81	58.30
$\mathcal{C}_5$	88.78	85.92
$\mathcal{C}_6$	95.88	96.12
Overall	89.18	87.16

port vector machine used in [18]. We note, however, the training-error convergence is rather slow when using the back propagation algorithm. On the contrary, the proposed methodology circumvents pre-processing and feature extraction making the process much faster. Even though the accuracy trade-off is about 6%, our method has the advantage in that it is more pragmatic for real-time applications where the data arrives sequentially one-by-one.

V. Conclusions

A methodology for human activity recognition was proposed in this paper. A committee of three feedforward neural networks was used to classify six typical human activities. With no pre-processing and feature extraction and with online sequential learning, the achieved classification accuracy is comparable to that of other techniques that rely on feature extraction. The proposed methodology is closer to the human cognitive process. The networks are trained using an alternative sequential variant of the extreme learning machine.

References

[1] J. L. Reyes-Ortiz, A. Ghio, D. Anguita, X. Parra, J. Cabestany, and A. Catala, "Human activity and motion disorder recognition: Towards smarter interactive cognitive environments," in *Proceedings of the 21st European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2013)*, Bruges, Belgium, April 2013, pp. 403–412.

[2] R. Poppe, "Vision-based human motion analysis," *Computer Vision and Image Understanding*, vol. 108, no. 1–2, pp. 4–18, 2007.

[3] D. M. Karantonis, M. R. Narayanan, M. Mathie, N. H. Lovell, and B. G. Celler, "Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring," *IEEE Transactions on Information Technology in Biomedicine*, vol. 10, no. 1, pp. 156–167, 2006.

[4] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal, "Histograms of oriented optical flow and Binet-Cauchy kernels on nonlinear dynamical systems for the recognition of human actions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, Miami, FL, USA, June 2009, pp. 1932–1939.

[5] F. I. Bashir, A. A. Khokhar, and D. Schonfeld, "Object trajectory-based activity classification and recognition using hidden markov models," *IEEE Transactions on Image Processing*, vol. 16, pp. 1912–1919, June 2007.

[6] D. Wyatt, M. Philipose, and T. Choudhury, "Unsupervised activity recognition using automatically mined common sense," in *Proceedings of the National Conference on Artificial Intelligence*, vol. 20, October 2005, pp. 21–27.

[7] 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*, October 2008, pp. 81–88.

[8] A. Khan, Y. K. Lee, S. 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*, May 2010, pp. 1–6.

[9] 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 *Proceedings of the 4th International Workshop on Ambient Assisted Living (IWAAL 2012)*, Vitoria-Gasteiz, Spain, December 2012, pp. 216–223.

[10] M. Ermes, J. Parkka, and L. Cluitmans, "Advancing from offline to online activity recognition with wearable sensors," in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, August 2008, pp. 4451–4454.

[11] B. Coley, B. Najafi, A. Paraschiv-Ionescu, and K. Aminian, "Stair climbing detection during daily physical activity using a miniature gyroscope. gait & posture," *Gait and Posture, Journal of Orthopaedic Science*, vol. 22, pp. 287–294, December 2005.

[12] 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*, vol. 22, October 2007, pp. 37–40.

[13] 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*, May 2009, pp. 2154–2159.

[14] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. Reyes-Ortiz, "Energy efficient smartphone-based activity recognition using fixed-point arithmetic," *Journal of Universal Computer Science*, vol. 19, pp. 1295–1314, May 2013.

[15] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: A new learning scheme of feedforward neural networks," in *Proceedings of IEEE International Joint Conference on Neural Networks (IJCNN2004)*, Budapest, Hungary, July 2004, pp. 985–990.

[16] 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.

[17] 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.

[18] 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.

[19] 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.

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