Deep Learning for Human Activity Recognition: A Resource Efficient Implementation on Low-Power Devices

Daniele Ravì, Charence Wong, Benny Lo and Guang-Zhong Yang

Abstract—Human Activity Recognition provides valuable contextual information for wellbeing, healthcare, and sport applications. Over the past decades, many machine learning approaches have been proposed to identify activities from inertial sensor data for specific applications. Most methods, however, are designed for offline processing rather than processing on the sensor node. In this paper, a human activity recognition technique based on a deep learning methodology is designed to enable accurate and real-time classification for low-power wearable devices. To obtain invariance against changes in sensor orientation, sensor placement, and in sensor acquisition rates, we design a feature generation process that is applied to the spectral domain of the inertial data. Specifically, the proposed method uses sums of temporal convolutions of the transformed input. Accuracy of the proposed approach is evaluated against the current state-of-the-art methods using both laboratory and real world activity datasets. A systematic analysis of the feature generation parameters and a comparison of activity recognition computation times on mobile devices and sensor nodes are also presented.

Index Terms—Deep Learning, Low-Power Devices, HAR, ActiveMiles

Fig. 1. Screenshot of ActiveMiles running on a LG Nexus 5 smartphone – our Android app for recording real world daily human activities [9].

I. INTRODUCTION

In wellbeing, healthcare, and sports monitoring, it is often important to capture the activities being performed by the subject. A detailed recording of the daily activities performed by the subject is important for many reasons, such as better understanding of wellbeing, knowing the activity being undertaken by a patient for diagnosis of pathologies, or identifying training activities for athletes. Fig. 1 shows a sample of the output generated from the smartphone application used to capture daily activity in this paper. Manual recording and labelling of different types of daily activity is labour intensive and infeasible for large population studies. The ability to accurately recognise different human activities automatically has significant implications for these applications.

For many years, machine learning and pattern recognition techniques for sensor-based classification of human activity have been focused on the design and use of "shallow" features that are task dependent. Features, such as mean [1], Fourier transforms [2], and symbols [3], are typically extracted from segments of data and then trained using classification methods [4]–[8]. However, these methods are still limited to the specific classification tasks that they were designed for.

Recently, there has been extensive interest in deep learning approaches for a diverse range of problems including
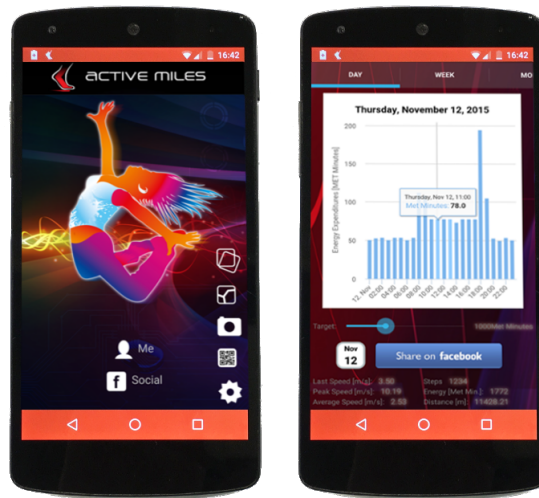
The Hamlyn Centre, Imperial College London, London {d.ravi, charence, benny.lo, g.z.yang}@imperial.ac.uk

image analysis, artificial intelligence, and sensor informatics. In deep learning, features are abstracted automatically from the data instead of being handcrafted, which allows these machine learning methods to be more effectively used across a range of different classification tasks [10]–[13]. This automatic feature extraction paradigm is also becoming increasingly relevant in the area of Body Sensor Networks (BSN) [14] as sensors are able to generate ever-growing amounts of data, which makes developing hand-crafted features a challenging task. In general, deep learning approaches work by using a hierarchy of layers that are initialised with randomly generated features. These features are gradually refined during the learning phase by optimising a cost function related to the classification task. The majority of deep learning approaches that have been proposed for Human Activity Recognition (HAR) attempt to extract discriminative features from the raw inertial sensor data. Changes in sensor orientation, sensor placement on the body, and other variations in sensor configuration make extracting descriptive features from raw data difficult to achieve. Where the input domain is not suitable for deep learning, current deep learning methods address this by exploiting a larger number of layers and nodes, resulting in higher complexity in the model, which is not an ideal solution for BSN. In this paper, we propose to combine the use of a suitable input domain with a well designed deep learning approach in order to obtain efficient classification features that are robust against transformations and variations in sensor properties.

The rest of the paper is organised as follows: In Section II, we introduce existing methods and the current state-of-the-art regarding deep learning approaches that have been proposed for HAR. Our proposed methodology for human activity recognition is described in Section III. Datasets used for evaluation and experimental results are presented in Section IV. The findings and contributions of our work are concluded in Section V.

II. RELATED WORK

Extracting discriminative features from the raw inertial data is a critical and challenging task for HAR. Most existing work relies on heuristic handcrafted features, also known as shallow features. Commonly used features for HAR include statistics of the raw signal (e.g. mean and variance of time sequences) [1], basis transform coding (e.g. signals with wavelet transform and Fourier transform) [2], and symbolic representation [3]. Classification methods, such as decision trees, k-Nearest Neighbour (k-NN), and Support Vector Machines (SVM), are then trained to identify different activities using the handcrafted features [4]–[6]. To further improve recognition accuracy, some researchers have demonstrated that ensemble classification methods, which combine multiple learning algorithms together, can achieve better outcomes in some cases. Catal et al. [7] and Zainudin et al. [8], for example, combine decision trees, multilayer perceptron, and logistic regression for HAR. Their results show that ensemble learning can obtain significant improvements for activity recognition when compared to what each learning algorithm can achieve individually with shallow features.

Deep learning approaches, such as Deep Belief Networks (DBN), Restricted Boltzmann Machines (RBM), and Convolutional Neural Networks (CNN), are being explored to create more generalised learning methods that extract features directly from the input data. However, current methods typically achieve this by introducing additional layers and nodes for classification, which increases computational complexity. For example, in the CNN based method described by Zeng et al. [10], additional max-pooling layers are applied after feature detection from the raw input to produce scale-invariant features, which is then introduced to a 1024 neuron hidden layer to merge features from multiple channels, and another additional soft-max layer is used to generate the classification result. Yang et al. [11] and Chen and Xue [12] both use CNNs with multiple iterations of convolution and subsampling layers, or convolution and pooling layers being applied for feature extraction. Alsheikh et al. [13] demonstrates activity recognition using a method based on DBNs. Deep Belief Networks are formed using multiple hidden layers, which in Alsheikh et al.'s implementation are formed from stacks of RBMs. A hybrid deep learning and hidden Markov model approach is finally used with three 1000 neuron layers. While utilising additional hidden layers and neurons to improve recognition accuracy is not a significant problem for high performance computer systems, it makes these methods unsuitable for devices with fewer resources, such as mobile devices and Body Sensor Networks.

III. METHODS

When a solution for the HAR problem is proposed the following considerations need to be addressed:

1) Raw inertial data is highly fluctuating over time, making classification impossible when using a single data point in time [15]. To tackle this issue, segments of finite length are usually generated and analysed at regular intervals. In this way, a sliding window strategy is adopted to segment the time series signal into a collection of short sequences of signals.
2) In many applications, there are no clues about the type or the position of the inertial sensor. Some approaches analyse the behaviour of human activities under a controlled environment (fixed orientation, position, sample rate, etc.). However, in real world scenarios, it is not always possible to fix these properties. For example, a HAR app for a smartphone can be executed on many different compatible devices, but no information about the type of sensor or the position/orientation where the smartphone is carried are available.

Due to the wide range of sensors available, it is challenging to develop an algorithm that works universally and we believe that generalisations of real word scenarios using handcrafted features is infeasible. Exploiting the idea of deep learning, our approach uses a systematic feature learning method for HAR. Our method adopts an efficient strategy to extract features using a temporal convolution on the spectrogram domain of the inertial data so that the learned features are invariant to changes in different properties. The rest of this section is divided as follows: In III-A, we first explain the spectrogram domain and its advantages for the HAR problem. In III-B, the details of our deep learning architecture are provided. III-C describes the training procedure and its regularisation, and finally in III-D, some information about the hardware and frameworks used to implement the proposed approach on low-power devices are given.

A. Spectrogram

A spectrogram of an inertial signal x is a new representation of the signal as a function of frequency and time. The spectrogram is the magnitude squared of the short-time Fourier transform (STFT). STFT is used to determine the sinusoidal frequency and phase content of local sections of a signal as it changes over time [16]. The procedure for computing the spectrogram is to divide a longer time signal into shorter segments of equal length and then compute the Fourier transform separately on each shorter segment. This reveals the Fourier spectrum on each shorter segment. The spectrogram describes the changing spectra as a function of time. In the discrete time case, the data to be transformed could be broken up into chunks or frames, which usually overlap each other to reduce artefacts at the boundaries. Each chunk is Fourier transformed, and the complex result is added to a matrix, which records magnitude and phase for each
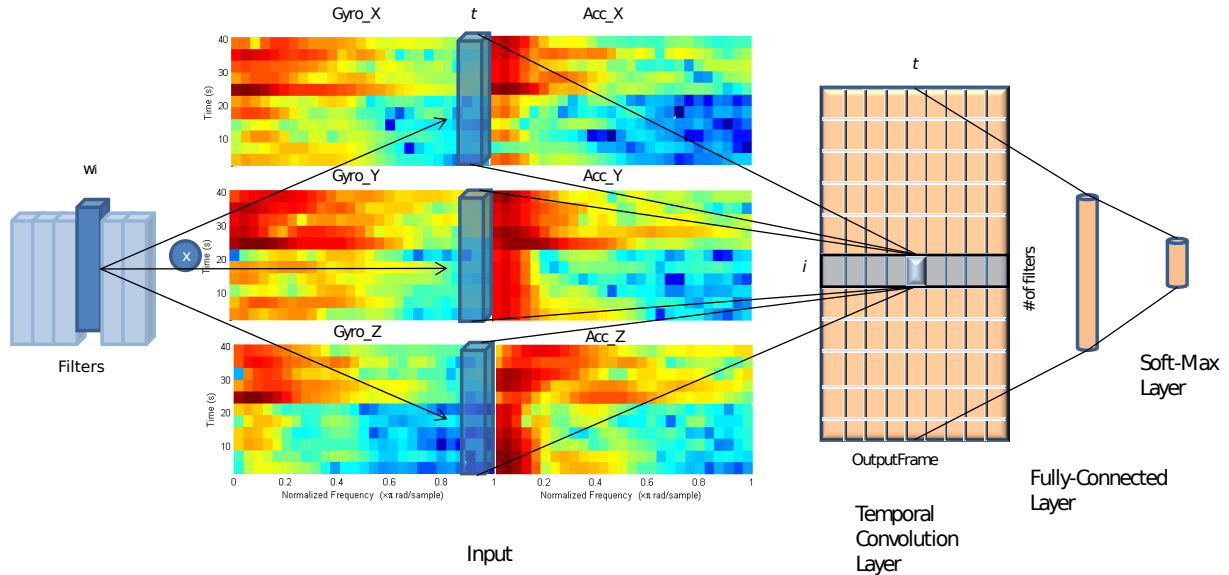
Fig. 2. Proposed deep learning architecture: a filter w_i is applied to the pre-arranged spectrograms of the input, and the weighted sums of the convolved signal at each time t are computed in the temporal convolution layer. Finally, the fully-connected layer and soft-max layer are used for classification.

point in time and frequency. This can be expressed as:

$$STFT\{x[n]\}(m,\omega) = X(m,\omega) = \sum_{n=-\infty}^{\infty} x[n]\omega[n-m]e^{-j\omega n} \quad (1)$$

likewise, with signal $x[n]$ and window $w[n]$. The magnitude squared of the STFT yields the spectrogram of the function:

$$spectrogram\{x(n)\}(m,\omega) = |X(m,\omega)|^2 \quad (2)$$

The resulting spectrogram is a matrix $st \times sf$ where st is the number of different short term, time-localised points and sf is the number of frequencies considered. Time increases across the columns while the frequency increases down the rows, starting at 0. The spectrogram representation provides interpretable features in capturing the intensity differences among nearest inertial data points. It also provides a form of temporal and sample rate invariance. This enables the classification of activities based on the variations of spectral density, which reduces classification complexity. We use the spectrogram representation as the input for the next stage of our deep leaning activity recognition model.

B. Feature Extraction using Deep Learning

In a typical deep learning approach for HAR, the time-series data obtained by the accelerometer and gyroscope are forwarded directly into the deep learning infrastructure. The deep learning model contains hidden relations between pairs of input signals; these correlations are usually overlooked. Moreover, a large set of layers are built on top of each other for automating feature design by trying to capture all the possible permutations of the signal using a greater number of nodes. Each layer performs a non-linear transformation on the outputs of the previous layer, so that the deep learning models the data by a hierarchy of features from low-level to high-level. These approaches have high

computation demands and make them unsuitable for low-power devices. In our approach we reduce the computation cost by limiting the connections from the input nodes, finding features efficiently through few nodes and levels. Our explicit connections are implemented using sums of local temporal convolutions over the transformed input. Since each human activity has a discriminative distribution of frequencies, the sum is performed in correspondence to each frequency obtained at different time-localised values. The implementation details of the proposed approach are provided in Algorithm 1. Specifically, each filter w_i – with size $kw \times st$ – is applied to the spectrogram vertically, as shown in Fig. 2, and the weighted sum of the convolved signal at time t is computed as follows:

$$o[t][i] = \sum_{j=1}^{st} \sum_{k=1}^{kw} w[i][j][k] * input[dw * (t-1) + k][j] \quad (3)$$

where dw is the stride of the temporal convolution.

These temporal convolutions produce an output layer o having size $wp \times OutputFrame$ with $OutputFrame = (InputFrame - kw)/dw + 1$ and wp the number of filters.

To introduce orientation invariance, temporal convolution results obtained from the different axes of an inertial sensor are integrated together without any discrimination, as shown in the for loop on Line 5 of Algorithm 1. Moreover, the filters applied to the three axes share the same weights, which is important for reducing the number of nodes in the layer and for better generalisation of the data. The last two layers of the proposed solution are a fully-connected layer and a soft-max layer, which are required since our deep model learns not only the informative features to recognise different activities from the data, but also the classification weights used to recognise the class activity.

Algorithm 1 Proposed algorithm for HAR

Input:

Acc_X, Acc_Y, Acc_Z ▷ Raw Triaxial Acceleration data
Gyr_X, Gyr_Y, Gyr_Z ▷ Raw Triaxial Gyroscope data
kw ▷ Kernel size for temporal convolution
sf ▷ Number of frequencies points in the spectrogram
st ▷ Number of time-localized points in the spectrogram
dw ▷ Step of the convolution
wp ▷ Number of filters

Output:

$result$ ▷ HAR label

1: $S\_a[1], S\_a[2], S\_a[3] \leftarrow spectr(Acc_X), spectr(Acc_Y), spectr(Acc_Z);$
2: $S\_g[1], S\_g[2], S\_g[3] \leftarrow spectr(Gyr_X), spectr(Gyr_Y), spectr(Gyr_Z);$
3: **for** i=1 to wp **do**
4:     **for** t=1 to sf **do**
5:         **for** z=1 to 3 **do**
6:             $o[t][i] += \sum\limits_{j=1}^{st} \sum\limits_{k=1}^{kw} w[i][j][k] * S\_g[z][dw*(t-1)+k][j];$
7:             $o[t+sf][i] += \sum\limits_{j=1}^{st} \sum\limits_{k=1}^{kw} w[i][j][k] * S\_a[z][dw*(t-1)+k][j];$
8:         **end for**
9:     **end for**
10: **end for**
11: $fc_o \leftarrow fully\_connected\_NN(o);$
12: $result \leftarrow soft\_max(fc_o);$

C. Training Process

For training our deep learning architecture, the inertial data extracted from fixed windows are converted to the spectral domain and forwarded in the network using a mini-batch of size equal to five. The forward propagation is conducted in the temporal convolutional layer using Eq. 3. The output of the convolutional layer is fed into the fully-connected layer, and then into the soft-max classifier, Fig. 2. After the first iteration of forward propagation has completed, we use the error value, obtained by the loss function L2-norm, in the backward propagation routine to update each edge of the network through the Stochastic Gradient Descent (SGD) approach. Moreover, to improve the training procedure of the weights, we have used three regularisations:

1) Weight Decay: Is an additional term in the weight update rule that causes the weights to exponentially decay to zero if no other update is scheduled. It is used to avoid over-fitting.

2) Momentum: Is a technique for accelerating gradient descent and attempting to move the global minimum of the function. It accumulates a velocity vector in directions of persistent reduction in the objective across iteration.

3) Dropout: Is a simple optimisation technique that removes units randomly from a neural network. Dropping out is done independently for each hidden unit and for each training case. Thus, applying dropout to a neural network is equal to sub-sampling a sub-neural network from it. Training a network with dropout leads to significantly lower generalisation error.

D. Implementation

The proposed approach has been deployed as an app for Android devices, Fig. 1, and also as an embedded algorithm
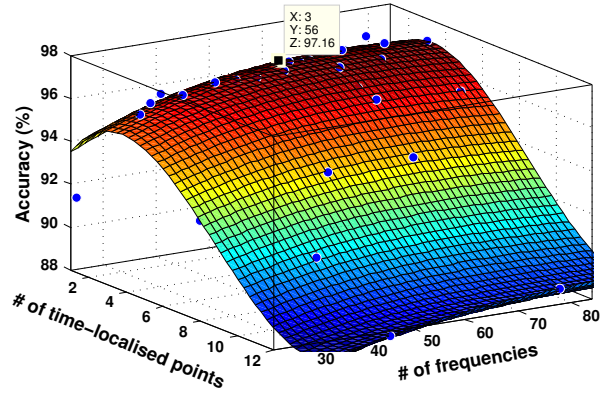

Fig. 3. Changes in classification accuracy on one of the considered datasets when the number of short time localised points for Fourier transform and the number of analysed frequencies are modified for spectrogram generation.

for the Intel Edison Development Platform. Intel Edison is utilised to demonstrate on-node human activity classification using the trained classification model. Featuring a dual-core Intel Atom CPU at 500MHz, wireless connectivity, and compact physical dimensions at $35.5 \times 25.0 \times 3.9$mm, the Intel Edison is a small but powerful platform that is well suited for smart body sensor networks. For both architectures, we use the FFTW3 library [17] to extract the spectrogram and the Torch framework [18] to implement the deep learning model. The spectrograms for each axis of each sensor (gyroscope and accelerometer) are computed separately using a fixed window size and chunks used to compute the Short-Time Fourier Transformations in Eq. 1 are obtained without overlap. The different spectrogram matrices are rearranged as shown in Fig. 2 and used as an input for the method.

IV. EXPERIMENTS & RESULTS

A. Datasets and Experimental Setup

To evaluate the performance of the proposed human activity recognition method on complex real world activity data and against existing approaches, four datasets are analysed using 10-fold cross validation. Table I summarises some of the properties of these datasets. Noteworthy is the release of our new dataset, ActiveMiles, which contains unconstrained real world human activity data from 10 subjects collected using five different smartphones. In addition to being the largest dataset in terms of number of samples, with around 30 hours of labelled raw data, as far as we know, it is the first database that contains data captured using different device configurations. To compare the relative performance of the proposed solution, we have used four different baseline configurations, as shown in Table II. After extracting 102 powerful handcrafted features collected from the most popular existing HAR approaches, we use a linear classifier and neural network to perform the classification task. A wrapper selection approach is also exploited with the aim to select the best 23 features [9]. In our experiments, we initially determine the optimal parameters for the proposed approach. Fig. 3 shows how the classification accuracy changes when

TABLE I

Dataset	Number of Activities	Subjects	Sensor Placement	Sensors		Sampling Rate	Samples	
				Accel.	Gyro.			
ActiveMiles	7	10	Any placement (unconstrained)	✓	✓	50 − 200Hz	4,390,726	
WISDM v1.1	6	29	Front trouser pocket (thigh)	✓	✗	20Hz	1,098,207	[19]
Daphnet FoG	2	10	Trunk, thigh, ankle	✓	✗	64Hz	1,917,887	[20]
Skoda	10	1	Arms (20 positions)	✓	✗	98Hz	∼ 701,440	[21]

TABLE II

BASELINE APPROACHES

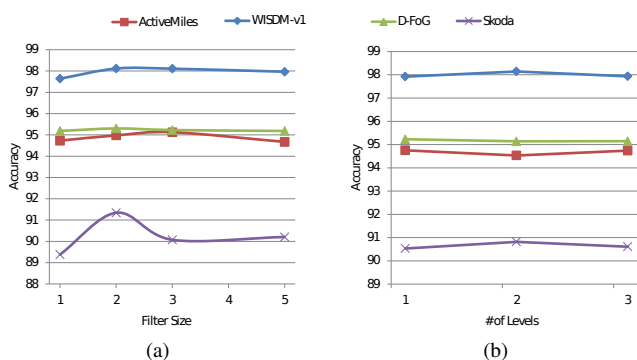| Name	Type of Features	# of Features	Classifier
Cs1	Wrapper Selection	23	Linear Classifier
Cs2	All Features	102	Linear Classifier
Cs3	Wrapper Selection	23	Neural Network
Cs4	All Features	102	Neural Network

Fig. 4. Accuracy of the proposed approach by increasing (b) the number of temporal convolution layers and (a) the size of temporal convolution kernel.
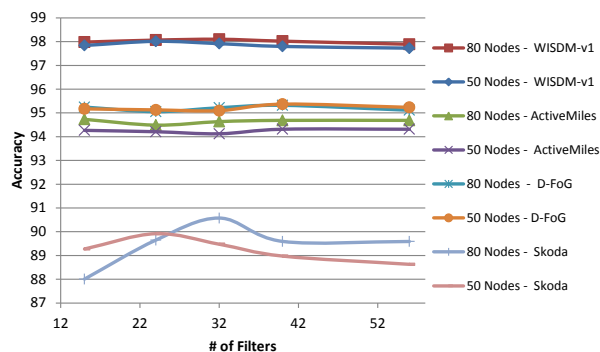

Fig. 5. Accuracy of the proposed approach against the number of filters used in the temporal convolution layer. For each dataset, 50 and 80 nodes are used in the fully connected layer.

the spectrogram generation parameters are modified. Specifically, two parameters are taken into account: The number of short time localised points for Fourier transform, and number of analysed frequencies. Selecting an appropriate number of short time localised points for the spectrogram is important, as it is evident from our analysis that too many points or too few points can compromise the accuracy. There is a gradual increase in the outcome when a greater number of analysed frequencies are used.

Fig. 4(b) shows that the proposed approach requires few levels in order to obtain good results. In our implementation, we use one temporal convolution layer since increasing the number does not produce substantial improvement. Also, the number of filters required for classification is minimal. Fig. 5 shows that 15 filters are enough and that just 80 nodes are required in the fully convolution layer. Fig. 4(a) shows that the optimal size of the temporal convolution kernel is two or three, depending on the data being classified.

A comparison of HAR results using the described baseline approaches, existing methods, and our proposed method are presented in Table III. Non-overlapping window sizes of 4s

and 10s were used to segment the time series of the raw inertial data. From the results shown, we can see that the proposed system generates discriminative features that are generally more powerful than the handcrafted features. However, in some cases using 102 handcrafted features together with a neural network produces marginally better results. For these particular tasks, the features learnt using our proposed method are not as discriminative as the handcrafted features, but our results are more consistent across all datasets. We obtain comparable accuracy results even against approaches that are more resource demanding [7], [13], [20], [21].

In Table IV, the computational time required to extract the spectrogram from the accelerometer and gyroscope sensors on 10 seconds of data, and the computation time for classification using the proposed deep learning approach are compared on two Android smartphones and on the Intel Edison Development Platform. Despite comparable specifications for the two smartphones tested, there is a slight difference in the computation time, which may be due to differences in operating system or other optimisations. For the implementation on the Intel Edison, we have recorded times using two different data sampling rates (50Hz and 200Hz), which is reflected in variation in spectrogram generation times. The times obtained from all platforms are consistent with the requirements for real-time HAR on low-power devices.

V. CONCLUSION & FUTURE WORKS

In this paper, we presented a deep learning method for human activity recognition and demonstrated that the classi-

TABLE III

COMPARISON OF OUR PROPOSED SOLUTION AGAINST EXISTING
METHODS IN DIFFERENT DATASET

Dataset	Approach	Window	Accuracy (%)	
ActiveMiles	Cs1		81.0	
	Cs2		89.3	
	Cs3	10s	94.7	
	Cs4		95.0	
	Ours		**95.1**	
WISDM v1.1	Cs1		85.1	
	Cs2		91.3	
	Cs3		96.7	
	Cs4	10s	97.4	
	[13]		**98.2**	
	[7]		94.3	
	Ours		**98.2**	
Skoda (Node 16)	Cs1		86.2	
	Cs2		89.2	
	Cs3		94.0	
	Cs4	4s	**95.9**	
	[21]		86.0	
	[13]		89.4	
	Ours		91.7	
			Sensitivity	Specificity
Daphnet FoG	Cs3		62.2	96.9
	Cs4		66.3	**97.7**
	[20]	4s	73.1	81.6
	[13]		**91.5**	91.5
	Ours		71.9	96.7

TABLE IV

COMPUTATION TIME FOR SPECTROGRAM GENERATION AND
CLASSIFICATION USING OUR METHOD ACROSS DIFFERENT PLATFORMS

Device	Spectrogram	Deep Learning
LG Nexus 5	5.4ms	5.7ms
Samsung Galaxy S5	20ms	8ms
Intel Edison	13ms – 42ms	14.9ms

fication stage could be implemented on low-power wearable devices.

To handle different sensor configurations and different types of activities, the raw inertial data is projected to the spectral domain. We have demonstrated that the feature generation process can create discriminative features that are more powerful than shallow handcrafted features. The accuracy of the proposed method is better or comparable against existing state-of-the-art approaches that utilise many more nodes and layers. The ability of the proposed method to generalise across different classification tasks is demonstrated using a variety of human activity datasets, including a dataset collected in unconstrained real world environments. Finally, we show that the computation times obtained from low-power devices, such as smartphones and BSN, are consistent with the requirements for real-time on-node human activity recognition.

REFERENCES

[1] A. Bulling, U. Blanke, and B. Schiele, "A Tutorial on Human Activity Recognition Using Body-worn Inertial Sensors," *ACM Comput. Surv.*, vol. 46, no. 3, Jan. 2014.

[2] T. Huynh and B. Schiele, "Analyzing Features for Activity Recognition," in *Proceedings of the 2005 Joint Conference on Smart Objects and Ambient Intelligence: Innovative Context-aware Services: Usages and Technologies*, ser. sOc-EUSAI '05. New York, NY, USA: ACM, 2005, pp. 159–163.

[3] J. Lin, E. Keogh, S. Lonardi, and B. Chiu, "A Symbolic Representation of Time Series, with Implications for Streaming Algorithms," in *Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery*, ser. DMKD '03. New York, NY, USA: ACM, 2003, pp. 2–11.

[4] Y. Lu, Y. Wei, L. Liu, J. Zhong, L. Sun, and Y. Liu, "Towards unsupervised physical activity recognition using smartphone accelerometers," pp. 1–19, 2016.

[5] A. Ignatov and V. Strijov, "Human activity recognition using quasiperiodic time series collected from a single tri-axial accelerometer," pp. 1–14, 2015.

[6] K. H. Walse, R. V. Dharaskar, and V. M. Thakare, "A study of human activity recognition using AdaBoost classifiers on WISDM dataset," *The Institute of Integrative Omics and Applied Biotechnology Journal*, vol. 7, no. 2, pp. 68–76, Jan. 2016.

[7] C. Catal, S. Tufekci, E. Pirmit, and G. Kocabag, "On the use of ensemble of classifiers for accelerometer-based activity recognition," *Applied Soft Computing*, vol. 37, pp. 1018–1022, Dec. 2015.

[8] M. N. S. Zainudin, M. N. Sulaiman, N. Mustapha, and T. Perumal, "Activity recognition based on accelerometer sensor using combinational classifiers," in *Open Systems (ICOS), 2015 IEEE Confernece on*. IEEE, Aug. 2015, pp. 68–73.

[9] D. Ravi, B. Lo, and G.-Z. Yang, "Real-time food intake classification and energy expenditure estimation on a mobile device," in *Wearable and Implantable Body Sensor Networks (BSN), 2015 IEEE 12th International Conference on*, Jun. 2015, pp. 1–6.

[10] M. Zeng, L. T. Nguyen, B. Yu, O. J. Mengshoel, J. Zhu, P. Wu, and J. Zhang, "Convolutional Neural Networks for human activity recognition using mobile sensors," in *Mobile Computing, Applications and Services (MobiCASE), 2014 6th International Conference on*. IEEE, Nov. 2014, pp. 197–205.

[11] J. B. Yang, M. N. Nguyen, P. P. San, X. L. Li, and S. Krishnaswamy, "Deep Convolutional Neural Networks on Multichannel Time Series for Human Activity Recognition," in *Proceedings of the 24th International Conference on Artificial Intelligence*, ser. IJCAI'15. AAAI Press, 2015, pp. 3995–4001.

[12] Y. Chen and Y. Xue, "A Deep Learning Approach to Human Activity Recognition Based on Single Accelerometer," in *Systems, Man, and Cybernetics (SMC), 2015 IEEE International Conference on*. IEEE, Oct. 2015, pp. 1488–1492.

[13] M. A. Alsheikh, A. Selim, D. Niyato, L. Doyle, S. Lin, and H.-P. Tan, "Deep Activity Recognition Models with Triaxial Accelerometers," Nov. 2015.

[14] G.-Z. Yang, "Body Sensor Networks," 2nd Edition, Springer, 2014, ISBN 978-1-4471-6374-9.

[15] O. D. Lara and M. A. Labrador, "A Survey on Human Activity Recognition using Wearable Sensors," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1192–1209, 2013.

[16] E. Sejdić, I. Djurović, and J. Jiang, "Timefrequency feature representation using energy concentration: An overview of recent advances," *Digital Signal Processing*, vol. 19, no. 1, pp. 153–183, Jan. 2009.

[17] M. Frigo and S. G. Johnson, "The design and implementation of fftw3," *Proceedings of the IEEE*, vol. 93, no. 2, pp. 216–231, 2005.

[18] R. Collobert, S. Bengio, and J. Mariéthoz, "Torch: a modular machine learning software library," IDIAP, Tech. Rep., 2002.

[19] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity Recognition Using Cell Phone Accelerometers," *SIGKDD Explor. Newsl.*, vol. 12, no. 2, pp. 74–82, Mar. 2011.

[20] M. Bächlin, M. Plotnik, D. Roggen, I. Maidan, J. M. Hausdorff, N. Giladi, and G. Tröster, "Wearable Assistant for Parkinson's Disease Patients with the Freezing of Gait Symptom," *Trans. Info. Tech. Biomed.*, vol. 14, no. 2, pp. 436–446, Mar. 2010.

[21] P. Zappi, C. Lombriser, T. Stiefmeier, E. Farella, D. Roggen, L. Benini, and G. Tröster, "Activity Recognition from On-Body Sensors: Accuracy-Power Trade-Off by Dynamic Sensor Selection," in *Wireless Sensor Networks*, ser. Lecture Notes in Computer Science, R. Verdone, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, vol. 4913, ch. 2, pp. 17–33.