Extending the Battery Lifetime of Wearable Sensors with Embedded Machine Learning

Xenofon Fafoutis*, Letizia Marchegiani†, Atis Elsts*, James Pope*, Robert Piechocki* and Ian Craddock*
*Department of Electrical and Electronic Engineering
University of Bristol, UK
Email: {xenofon.fafoutis, atis.elsts, james.pope, r.j.piechocki, ian.craddock}@bristol.ac.uk
†Department of Engineering Science
University of Oxford, UK
Email: letizia.marchegiani@eng.ox.ac.uk

***Abstract*—Smart health home systems and assisted living architectures rely on severely energy-constrained sensing devices, such as wearable sensors, for the generation of data and their reliable wireless communication to a central location. However, the need for recharging the battery regularly constitutes a maintenance burden that hinders the long-term cost-effectiveness of these systems, especially for health-oriented applications that target people in need, such as the elderly or the chronically ill. These sensing systems generate raw data that is processed into knowledge by reasoning and machine learning algorithms. This paper investigates the benefits of embedded machine learning, i.e. executing this knowledge extraction on the wearable sensor, instead of communicating abundant raw data over the low power network. Focusing on a simple classification task and using an accelerometer-based wearable sensor, we demonstrate that embedded machine learning has the potential to reduce the radio and processor duty cycle by several orders of magnitude; and, thus, substantially extend the battery lifetime of resource-constrained wearable sensors.**

***Index Terms*—Wearable systems, Embedded Machine Learning, eHealth, Internet of Things (IoT)**

I. Introduction

Our health systems are challenged by rising trends in chronic illness and ageing populations. Yet, the emerging Internet of Things (IoT) along with advances in sensing technology and microelectronics, constitute a promising means for off-loading the medical sector and revolutionise healthcare provision. For instance, sensing technology is a key enabler of behavioural monitoring systems, which can collect detailed information about the long-term behavioural habits of their users - information that is otherwise difficult to obtain [1]. Such information can be then shared with clinicians and other medical professionals for quicker and more informed decision making. Moreover, sensing technology is the backbone of assisted living infrastructures, aiming at the prompt detection of emergencies and their cost-effective intervention [2].

Such eHealth infrastructures are complex systems that require multidisciplinary expertise to operate efficiently and be valuable to their users. On one end, there are embedded systems that are equipped with sensors that monitor the users (or their environment) and generate data. Such systems typically operate on very limited resources in terms of energy, memory and processing power. Running on a very limited energy budget, a sensing system must connect to a communication infrastructure, which typically manifests in the form of a wireless sensor network, and transfers the generated data to a centralised processing and storage unit, which may be hosted locally or in the cloud. Data collected from a number of distributed sensors can then be fused together, processed and, through reasoning techniques and machine learning algorithms, transformed into valuable knowledge. In the final step of the process, a number of services build upon the inferred knowledge aiming to offer some value to the user.

In order to effectively fulfil their purpose, sensing infrastructures must require little-to-no maintenance. In the case of battery-powered sensing devices, in particular, the primary maintenance cost originates from the need for regular battery charging or replacement. Currently, many commercial electronic gadgets, such as smart phones or activity trackers [3], depend on the user for this regular maintenance. However, this approach hinders the applicability of such devices for critical health-oriented applications. As an example consider that the cognitive effort of managing multiple devices can be unbearable for certain parts of the population, such as the elderly. As a result, over the years, there has been a tremendous effort from the research community to improve the energy efficiency of every part of these energy-constrained sensing systems: from energy-efficient sensing elements [4] to duty-cycling low power networks [5], and from energy-efficient security [6] to low-power operating systems [7].

Yet, a sensing system is only as efficient as its least efficient subsystem. In other words, the tremendous achievements of decades of research in low power systems and networks would not be reflected on the overall system performance, unless the principles of resource-efficient design are also adopted in the data layers. Indeed, no matter how energy-efficient the sensing system becomes, its energy requirements also depend on the amount of data that it has to handle. Therefore, any data that is generated, transferred, stored, or processed unnecessarily is a potential waste of precious energy. This paper builds on this principle, focusing on improving the battery lifetime of

978-1-5090-4130-5/16/$31.00 ©2018 IEEE

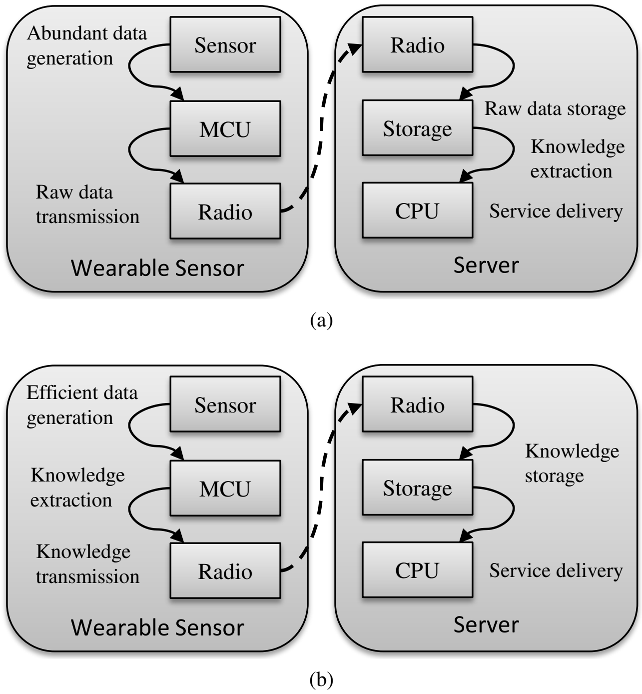

Fig. 1. The raw data approach (a): Abundant data is generated and sent over the air from the wearable sensor to a central server for long-term storage and post-processing. The embedded machine learning approach (b): The data generation is tailored to a particular application and the knowledge extraction is executed locally. Only the extracted knowledge is sent to the central server.

energy-constrained sensing devices with embedded machine learning. The key concept is that extracting knowledge from raw sensor data is a distillation process by definition, *i.e.* the output knowledge can be represented in fewer bytes than the original raw data. Therefore, unless it introduces unbearable processing overhead for embedded micro-controllers, it is beneficial for the battery lifetime of the device to execute the knowledge extraction as close to the data source as possible.

As a use case scenario, this paper considers the eHealth application of using a wearable sensor in a residential environment to identify the physical activity levels of a user and classify them into three categories: sedentary, moderate, and vigorous activities. Due to their size and weight constraints, wearable sensors are equipped with tiny batteries, and, thus, are severely energy-constrained [8]. The paper is then organised as follows. After a brief dicussion of the related work (Section II), we design the application following a raw data approach (illustrated in Fig. 1a), providing insight about the classification performance and energy costs that are required for achieving it (Section III). Then, we gradually investigate the benefits of embedded machine learning (illustrated in Fig. 1b), optimising the data flow (from data generation to knowledge extraction) and moving the knowledge extraction in the embedded device (Section IV). In Section V, we then discuss the limitations and trade-offs of embedded machine learning. Lastly, Section VI concludes the paper.

II. Related Work

Most of the machine learning systems are not targeted for embedded systems. Exceptions can be found in the fields of robotics [9], imaging [10] and computer vision [11]. Yet, compared with the field of IoT, these embedded applications typically operate on less resource-constrained hardware. In the field of wearable computing, more specifically, embedded classification and hardware-accelerated machine learning have been used for EEG (electroencephalogram) and ECG (electrocardiogram) signals [12], [13].

Targeting such embedded applications, the literature also includes works that focus on optimising a specific classifier for embedded hardware. Examples can be found for Support Vector Machines (SVM) [14], artificial neural networks [15] and deep learning [16]. In addition, there have been efforts on machine learning algorithms that reduce the amount of data that needs to be stored on the target hardware, focusing on embedded devices with memory constraints [17].

Fundamentally, the designer of a knowledge extraction system that builds on data that originates from resource-constrained sensing systems needs to balance the trade-off between the accuracy of the output knowledge and the cost of collecting the data. This trade-off is reported in the literature as the cost-accuracy conflict [18]. A number of works attempt to solve this conflict proposing the assignment of a cost value to each potential feature. Hence, the goal of the learning process is to jointly minimise both the cost and classification error. This cost value can be an abstract measure [19], or it can depend on computational costs [20] or financial costs [21].

Different from the works that are briefly presented in this section, the purpose of this paper is to investigate the benefits of embedded machine learning from a full-system perspective. Focusing on energy-constrained devices, our interest is in identifying and quantifying the potential benefits (in terms of battery lifetime) of using embedded machine learning as a means to reduce the radio duty cycle of the processor and the radio, *i.e.* the typically two most energy-consuming components of a low-power IoT device.

III. Classification of Physical Activity Levels

Let us consider a residential monitoring system that makes use of wearable sensors to track the physical activity levels of house occupants. The goal of the system is to periodically classify the activity levels of the users into three categories: sedentary, moderate, and vigorous activities.

A. Data Collection

For the purposes of this work, we use the SPHERE Wearable sensor [22], shown in Fig. 2. The SPHERE wearable is one of the three key sensing modalities of SPHERE (a Sensor Platform for HEalthcare in a Residential Environment), a multi-modal sensing infrastructure for long-term behavioural monitoring for healthcare-oriented purposes. The SPHERE wearable is a wrist-mounted accelerometer-based activity sensor that is designed for long-term residential monitoring with minimum maintenance. Its first generation is based on the

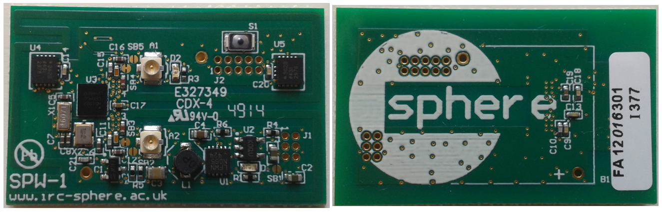

Fig. 2. The SPHERE wearable used for the experimental results [22]. Top view (left) and bottom view (right).

nRF51822 System-on-Chip (SoC) that incorporates a Cortex M0 processor and a BLE (Bluetooth Low Energy) radio. The board is equipped with the ADXL362, a digital triaxial accelerometer that can support up to 400 Hz sampling frequency and 12 bits of resolution (*i.e.* 2 mg resolution at ± 4 g acceleration range) [23].

We use the SPHERE wearable to collect a labelled dataset of the acceleration profile of six different activities of daily life. In particular, we consider two sedentary activities (watching TV and typing on a computer), two moderate activities (walking and housework) and two vigorous activities (running and exercising). For each one of the six activities, we collect the acceleration profile of 10 different instances (a total number of 60 acceleration profiles). Moreover, each activity is executed in different ways (for example, walking straight, walking upwards, walking downwards, etc.). The activities are summarised in Table I. For the data collection, we configure the accelerometer to output acceleration samples of 12-bit resolution at 50 Hz sampling frequency. Indeed, 50-100 Hz sampling frequency is a common configuration in the activity recognition literature, as it is abundant enough for a wide range of activities [24].

TABLE I
LIST OF ACTIVITIES

Index	Class	Activity
0	Rigorous	Running
0	Rigorous	Exercising
1	Moderate	House Cleaning
1	Moderate	Walking
2	Sedentary	Typing
2	Sedentary	Watching TV

B. Feature Extraction and Classification

The classification process is using the Integral of the Modulus of Acceleration (`IMA`) as input feature. This metric is commonly used in the literature for estimating physical activity levels [25], as it correlates with the energy expenditure of a person [26]. The magnitude of a single acceleration sample, $\vec{a} \in \mathbb{R}^3$, is calculated as in [27]:

$$||\vec{a}|| = \left|\sqrt{a_x^2 + a_y^2 + a_z^2} - 1\right| \qquad (1)$$

where a_x, a_y and a_z denote the acceleration on each of the three axes respectively, measured in g-units ($g = 9.8\ m/s^2$). The gravity component ($1\ g$) is subtracted from the magnitude of the raw signal to isolate the acceleration of the user.

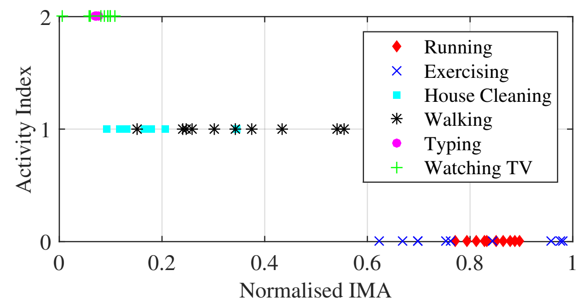

Fig. 3. Feature extraction for 60 labelled activities collected with the SPHERE wearable. The activity index (y-axis) corresponds to: sedentary (0), moderate (1) and vigorous (2) activities.

Considering a discrete time series of samples, $\vec{a}_0, \vec{a}_1 \ldots \vec{a}_n$, the normalised *IMA* of an acceleration profile of w samples is then approximated as:

$$\mathtt{IMA} = \frac{1}{w} \sum_{i=0}^{w-1} ||\vec{a}_i||. \qquad (2)$$

The `IMA` values of the collected activities are summarised in Fig. 3. The y-axis corresponds to the index of Table I.

For the classification we employ the SVM classifier. SVM is a supervised classifier that leverages labelled training data to identify a hyperplane that maximises the margin between two classes. Two different methods can be used to extend the SVM framework to a multi-class scenario. In this paper, we employ the one-versus-one approach, *i.e.* classification is obtained by combining classifiers between each pair of classes. Yet, given the nature of the data analysed, only two classifiers are necessary. Indeed, there will be no direct overlap between samples belonging to class 0 and class 2, as samples in class 1 lie in their separation margin. We use two SVM classifiers, based on the RBF (Radial Basis Function) kernel (configured with a box constraint $c = 100$ and a scaling factor $\sigma = 1$). More specifically, we split the data into two random sets: the first is used for training (66%) the classifiers and the second is used for testing their accuracy (33%). Lastly, we repeat the process $10,000$ times on different random training and testing data sets.

The average classification accuracy is 93.23% with a standard deviation of 6.57%.

C. Battery Lifetime of the Wearable Device

The aforementioned procedure is following a raw data approach, illustrated in Fig. 1a. The raw data is generated by the sensor (*i.e.* the accelerometer) and transferred to the micro-controller unit (MCU) using an intra-board short-range communication protocol, in this case SPI (Serial Peripheral Interface). The MCU is then buffering and processing the data, packing them into BLE advertisements. Following the packet format of SPHERE [1], a BLE advertisement contains 18 bytes of raw data and 6 bytes of control and monitoring data, in addition to the headers of BLE. Hence, a BLE advertisement has sufficient space for 4 triaxial acceleration samples of 12 bits of resolution. The formatted BLE advertisements are then passed to the BLE radio which transmits the raw data over the smart home network to a central unit for processing and long-term storage. The feature extraction and classification follows. Effectively, this approach collects the data via a residential broadcast network that relies on non-connectable undirected BLE advertisements. Its performance, in terms of reliability, is investigated in [28].

A comprehensive energy measurement campaign of the SPHERE wearable sensor can be found in [29]. Using those measurements, together with the datasheet of the accelerometer [23], we next profile the energy requirements of the aforementioned procedure and estimate the battery lifetime of the wearable sensor.

The board has an idle current of approximately 8 μW, when all the components are in sleep mode [29]. The accelerometer itself is ultra-low-power, requiring roughly 3 μW for generating raw data at a sampling frequency of 50 Hz [23]. Transferring the raw data from the sensor to the MCU, however, consumes significantly more. The transfer of a single sample consumes approximately 1.9 μJ [29], resulting to 95 μW at 50 Hz. The wireless transmission is the biggest source of energy consumption. The preparation and transmission of an advertisement packet (+4 dBm transmission power) consumes approximately 61.7 μJ [29], which corresponds to 771 μW at 50 Hz. The long-term average power consumption of the wearable sensor is given by the sum of aforementioned elements, *i.e.* 887 μW. Assuming an energy budget of 1000 J (typical for the size of a wearable battery), this energy consumption rate corresponds to a battery lifetime of approximately 13 days.

IV. Embedded Machine Learning

With the raw data approach, the application enjoys an accuracy rate of 93.23%, yet at the maintenance cost of recharging the wearable sensor once every roughly two weeks. In this section, we explore the embedded machine learning approach (illustrated in Fig. 1b) in an attempt to extend the battery lifetime of the wearable sensor without compromising the classification accuracy. To this end, we follow the principle that any data that is generated, transferred, stored, or processed unnecessarily is a potential waste of energy. As a first step, we focus on data generation and optimise the sensing process. The following step is moving the machine learning elements (feature extraction and classification) in the embedded device.

A. Sampling

There is a plethora of works that operate on potentially redundant input data, without focusing on the cost of obtaining these data (*e.g.* [30], [31]). Khan *et al.* [32] investigated this inefficiency in the context of accelerometer-based human activity recognition and concluded that the sampling rates that are used in the literature are up to 57% higher than what is needed, leading to the waste of precious resources.

In this section we explore and quantify the benefits of optimising the data generation to the needs of our application of interest. To this end, we explore two dimensions: not only the sampling frequency, but also the sampling resolution. In particular, we start with the raw data collected in Section III-A and we gradually drop samples (effectively dividing the sampling frequency by a factor of two), as well as bits of resolution starting from the least significant bit. With each new reduced dataset, we then repeat the machine learning procedure (training and testing), exactly as described in Section III-B. The results are summarised in Fig. 4, in which the x-axis corresponds to the bit resolution and the y-axis corresponds to the sample frequency. The classification accuracy of each sampling configuration is colour coded. It can be observed that the classification task can be effectively executed ($> 90\%$) with a sampling frequency of as low as 0.39 Hz and a bit resolution of 5 bits. This sampling configuration reduces the amount of generated data by 3 orders of magnitude without compromising the classification performance.

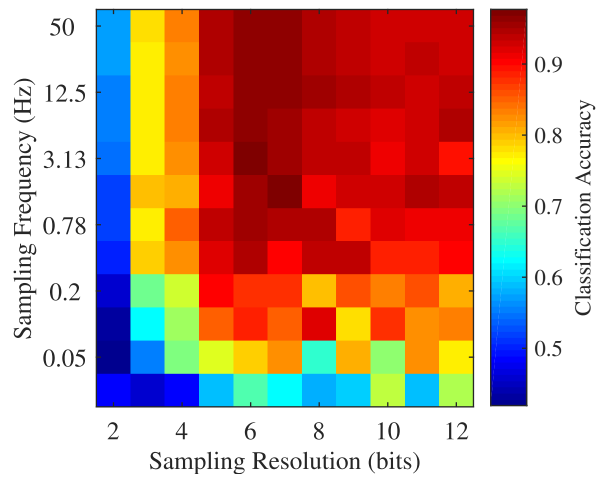

Fig. 4. The classification accuracy under various data collection strategies in terms of sampling frequency (y-axis) and sample resolution (x-axis).

In practice, it is impossible to implement certain sampling configurations due to hardware limitations. In the case of the SPHERE wearable, the ADXL362 accelerometer has a fixed sampling resolution that cannot be altered (12 bits). However, the process of transferring the samples from the accelerometer to the MCU can be improved. Indeed, the 12-bit samples are stored in two 8-bit registers. By reading and transferring the most significant byte only, the cost of transferring the data to the MCU can be reduced by a factor of two (0.95 μJ per sample), resulting to an effective sample resolution that is sufficiently high, *i.e.* 8 bits.

With regard to the sampling frequency, the lowest configuration supported by ADXL362 is 12.5 Hz. This, however, has a small effect on the energy requirements of sensing, *i.e.* less than 0.5 μW [23]. Nevertheless, significant amounts of energy can be saved by configuring the MCU to poll the accelerometer at a lower frequency. Indeed polling 8-bit accelerometer samples (0.95 μJ per sample) corresponds to 0.37 μW at 0.39 Hz. This is a reduction of the cost of transferring samples within the board by 3 orders of magnitude, as opposed to the original configuration (*i.e.* 95 μW, see Section III-C).

At this stage, there are two design options. The first is to pass the data to the BLE radio for transmission. With 8-bit samples, the system can now pack 6 triaxial acceleration samples in a BLE advertisement. With the energy cost of 61.7 μJ per advertisement, the transmission of the data to the smart house requires 4 μW at the reduced sampling frequency of 0.39 Hz. In this configuration, the long-term average power consumption of the wearable sensor is approximately 15 μW. Assuming an energy budget of 1000 J, this energy consumption rate corresponds to a battery lifetime of approximately 771 days. The alternative option is to execute the feature extraction and classification in the embedded system.

B. Embedded Feature Extraction and Classification

Embedded feature extraction can be a double-edged sword. On one hand, it can reduce the amount of data that needs to be transferred over the wireless medium, effectively reducing the duty cycle of the radio. On the other hand, it introduces additional processing costs in the embedded system. Extracting different types of features can tip the scale one way or the other. Therefore, whether embedded feature extraction is beneficial or not, must be investigated on a per-case basis.

In the particular application investigated in this paper, the level of data reduction depends on the window w. Indeed, upon the execution of equations (1) and (2), the output data is smaller than the input data by a factor of $\frac{2}{3w}$, where the factor $\frac{1}{3}$ is because of (1) that compresses the 3 dimensions into their magnitude, and the factor 2 is because of the fact that `IMA` requires a 16-bit variable in the worst case scenario. Hence, the cost of wireless transmission is reduced respectively, *i.e.* $\frac{2.67}{w}$ μW.

Next, we implement a function that executes the equations (1) and (2), and we measure the energy required for its execution by the processor of the wearable sensor, using the exact same setup as in [29]. In particular, we use a 10 Ω resistor in series with the power supply and we measure the voltage drop across it. Using this setup, we measure that extracting the `IMA` feature requires 2.2 μJ per sample (that is approximately 0.56 μW at 0.39 Hz). In other words, the window needs to be $w > 4$ for embedded feature extraction to be beneficial to the energy efficiency of the wearable system.

Our implementation uses a window of $w = 32$ that corresponds to approximately 80 seconds. Note that it is advised to use a window that is a power of 2, as this enables efficient divisions by shifting. In this configuration, the long-term average power consumption of the wearable system is the sum of the idle power (8 μW), the average power consumed for generating and transferring the data from the sensor to the MCU (3 μW), the cost for the feature extraction (0.56 μW), and the cost for wireless transmission (0.09 μW). This sums to approximately 11.7 μW. Assuming an energy budget of 1000 J, this energy consumption rate corresponds to a battery lifetime of approximately 989 days.

Embedded classification is then implemented as a simple decision tree with two `if` statements that test the extracted feature against the two `IMA` thresholds that separate the sedentary from the moderate activities and the moderate from the vigorous activities. These thresholds are 0.1184 and 0.5931 respectively, derived from the SVM model. The predicted activity index can, then, be encoded in 2 bits. This further reduces the cost of wireless transmission by a factor of 8 (0.01 μW). This sums to an average long-term power consumption of approximately 11.6 μW. Given an energy budget of 1000 J, this energy consumption rate corresponds to a battery lifetime of approximately 997 days. Although this stage has reduced the radio duty cycle by almost an order of magnitude, the overall improvement is indeed small. This is due to the idle power that has become the performance bottleneck and highlights the need for more energy-efficient components on the wearable sensor.

TABLE II
SUMMARY OF RESULTS

	Average Power	Battery Lifetime
Raw data collection	887 μW	13 days
Optimisation of sampling	15 μW	771 days
Embedded feature extraction	11.7 μW	989 days
Embedded classification	11.6 μW	997 days

The results are summarised in Table II. It can be observed that the reduction of the energy costs of the data flow has diminishing returns due to the idle consumption that gradually becomes the dominant factor of power consumption. Nevertheless, with embedded machine learning the battery lifetime of the wearable sensor is increased from weeks to years.

V. DISCUSSION

Despite the potentially very substantial improvements in the battery lifetime of resource-constrained sensors and, thus, in the maintenance overhead of the eHealth sensing system, embedded machine learning has its own cost. Indeed, collecting and storing abundant raw data has the following advantages that are hindered by the embedded machine learning approach. Firstly, abundant raw data can potentially enable multiple applications, whereas optimising the data generation (sampling frequency and resolution) to a particular task leads to specialised systems. Moreover, the raw data approach effectively decouples the data collection from the knowledge extraction, leading to application-agnostic systems that are more future-proof by design [33]. Fundamentally, this is trade-off between efficiency and versatility.

In the context of research data in particular, it should also be mentioned that raw data enables the validation of research results by third parties (reproducible research). For this reason, raw data collection, storage and publication may be enforced by publishers and funding agencies [34].

With no doubts, the raw data approach has important advantages that should not be neglected. Instead, the designers of eHealth monitoring systems should weight the advantages of both approaches and decide according to their design priorities.

VI. Conclusions

Using the classification of physical activity levels as a use case scenario, this paper investigates the benefits of using embedded machine learning as a means of extending the battery lifetime of severely energy-constrained embedded systems, such as wearable sensors. By optimising the data collection to a particular application and moving the knowledge extraction closer to the data source, we have reduced the duty cycle of the radio and processor by several orders of magnitude, effectively extending the battery lifetime of the wearable sensor from weeks to years. These results highlight the fact that the designer of an eHealth monitoring system has the option to sacrifice the versatility offered by collecting raw data for substantial reductions on the maintenance cost of severely constrained devices that are used for long-term monitoring.

Acknowledgment

This work was performed under the SPHERE IRC funded by the UK Engineering and Physical Sciences Research Council (EPSRC), Grant EP/K031910/1. The authors would also like to thank Yaojia Cui for assisting with data collection.

References

[1] P. Woznowski, A. Burrows, T. Diethe *et al.*, "SPHERE: A Sensor Platform for Healthcare in a Residential Environment," in *Designing, Developing, and Facilitating Smart Cities: Urban Design to IoT Solutions*. Springer International Publishing, 2017, pp. 315–333.

[2] P. Rashidi and A. Mihailidis, "A survey on ambient-assisted living tools for older adults," *IEEE J. Biomed. Health Inform*, vol. 17, no. 3, pp. 579–590, 2013.

[3] T. J. M. Kooiman, M. L. Dontje, S. R. Sprenger, W. P. Krijnen, C. P. van der Schans, and M. de Groot, "Reliability and validity of ten consumer activity trackers." *BMC Sports Sci. Med. Rehabil.*, vol. 7, no. 1, p. 24, Jan. 2015.

[4] V. Kumar, R. Jafari, and S. Pourkamali, "Ultra-low power digitally operated tunable mems accelerometer," *IEEE Sensors J.*, vol. 16, no. 24, pp. 8715–8721, 2016.

[5] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados, "Energy-efficient routing protocols in wireless sensor networks: A survey," *Commun. Surveys Tuts*, vol. 15, no. 2, pp. 551–591, 2013.

[6] A. Di Mauro, X. Fafoutis, S. Mödersheim, and N. Dragoni, "Detecting and preventing beacon replay attacks in receiver-initiated mac protocols for energy efficient wsns," in *Proc. 18th Nordic Conf. Secure IT Systems, (NordSec)*, H. Riis Nielson and D. Gollmann, Eds., 2013, pp. 1–16.

[7] A. Dunkels, B. Gronvall, and T. Voigt, "Contiki - a lightweight and flexible operating system for tiny networked sensors," in *29th IEEE Int. Conf. on Local Computer Networks*, 2004, pp. 455–462.

[8] M. Chan, D. Estève, J.-Y. Fourniols, C. Escriba, and E. Campo, "Smart wearable systems: Current status and future challenges," *Artificial Intell. in Medicine*, vol. 56, no. 3, pp. 137 – 156, 2012.

[9] H. Guo, H. Xiao, S. Wang, W. He, and K. Yuan, "Real-time detection and classification of machine parts with embedded system for industrial robot grasping," in *IEEE Int. Conf. on Mechatronics and Automation (ICMA)*, 2015, pp. 1691–1696.

[10] J. Choi, S. Park, J. Cho, and E. Yoon, "A 3.4-μW Object-Adaptive CMOS Image Sensor With Embedded Feature Extraction Algorithm for Motion-Triggered Object-of-Interest Imaging," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 1, pp. 289–300, Jan 2014.

[11] F. Zuo and P. H. N. de With, "Real-time embedded face recognition for smart home," *IEEE Trans. Consum. Electron*, vol. 51, no. 1, pp. 183–190, Feb 2005.

[12] A. Shoeb, D. Carlson, E. Panken, and T. Denison, "A micropower support vector machine based seizure detection architecture for embedded medical devices," in *Int. Conf. IEEE Engineering in Medicine and Biology Society*, 2009, pp. 4202–4205.

[13] K. H. Lee and N. Verma, "A low-power processor with configurable embedded machine-learning accelerators for high-order and adaptive analysis of medical-sensor signals," *IEEE J. Solid-State Circuits*, vol. 48, no. 7, pp. 1625–1637, 2013.

[14] A. Boni, F. Pianegiani, and D. Petri, "Low-power and low-cost implementation of svms for smart sensors," *IEEE Trans. Instrum. Meas*, vol. 56, no. 1, pp. 39–44, 2007.

[15] N. N. Charniya, "Design of near-optimal classifier using multi-layer perceptron neural networks for intelligent sensors," *International Journal of Modeling and Optimization*, vol. 3, no. 1, pp. 56–60, 2013.

[16] D. Ravi, C. Wong, B. Lo, and G. Z. Yang, "Deep learning for human activity recognition: A resource efficient implementation on low-power devices," in *IEEE 13th Int. Conf. on Wearable and Implantable Body Sensor Networks (BSN)*, 2016, pp. 71–76.

[17] A. Jain and D. Nikovski, "Incremental exemplar learning schemes for classification on embedded devices," *Mach. Learn.*, vol. 72, no. 3, pp. 189–203, 2008.

[18] U. Jensen, P. Kugler, M. Ring, and B. M. Eskofier, "Approaching the accuracy–cost conflict in embedded classification system design," *Pattern Analysis and Applications*, vol. 19, no. 3, pp. 839–855, 2016.

[19] K. Iswandy and A. Koenig, "Feature selection with acquisition cost for optimizing sensor system design," *Advances in Radio Science*, vol. 4, pp. 135–141, 2006.

[20] M. Tan, "Cost-sensitive learning of classification knowledge and its applications in robotics," *Mach. Learn.*, vol. 13, no. 1, pp. 7–33, 1993.

[21] Y. Weiss, Y. Elovici, and L. Rokach, "The cash algorithm-cost-sensitive attribute selection using histograms," *Inf. Sci.*, vol. 222, pp. 247–268, 2013.

[22] X. Fafoutis, A. Vafeas, B. Janko *et al.*, "Designing Wearable Sensing Platforms for Healthcare in a Residential Environment," *EAI Endorsed Trans. Pervasive Health and Technology*, vol. 17, no. 12, Sept. 2017.

[23] Analog Devices, "ADXL362 - Micropower, 3-Axis, ± 2 g / ± 4 g / ± 8 g, Digital Output MEMS Accelerometer, Rev. B," [Online] Available at: http://www.analog.com/static/imported-files/data_sheets/ADXL362.pdf, 2013.

[24] O. D. Lara and M. A. Labrador, "A survey on human activity recognition using wearable sensors," *Commun. Surveys Tuts*, vol. 15, no. 3, pp. 1192–1209, 2013.

[25] S. Bosch, R. Marin-Perianu, P. Havinga, and M. Marin-Perianu, "Energy-Efficient Assessment of Physical Activity Level Using Duty-Cycled Accelerometer Data," *Procedia Computer Science*, vol. 5, pp. 328–335, 2011.

[26] C. V. Bouten, K. R. Westerterp, M. Verduin, and J. D. Janssen, "Assessment of energy expenditure for physical activity using a triaxial accelerometer." *Medicine and science in sports and exercise*, vol. 26, no. 12, pp. 1516–23, 1994.

[27] X. Fafoutis, L. Marchegiani, G. Z. Papadopoulos, R. Piechocki, T. Tryfonas, and G. Oikonomou, "Privacy leakage of physical activity levels in wireless embedded wearable systems," *IEEE Signal Process. Lett*, vol. 24, no. 2, pp. 136–140, 2017.

[28] E. Tsimbalo, X. Fafoutis, E. Mellios *et al.*, "Mitigating Packet Loss in Connectionless Bluetooth Low Energy," in *Proc. 2nd IEEE World Forum on Internet of Things (WF-IoT)*, 2015, pp. 291–296.

[29] X. Fafoutis, B. Janko, E. Mellios, G. Hilton, R. S. Sherratt, R. Piechocki, and I. Craddock, "SPW-1: A Low-Maintenance Wearable Activity Tracker for Residential Monitoring and Healthcare Applications," in *Proc. Int. Summit on eHealth (eHealth 360)*, 2016, pp. 294–305.

[30] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. Reyes-Ortiz, "A public domain dataset for human activity recognition using smartphones," in *European Symp. on Artificial Neural Networks, Computational Intell. and Mach. Learning (ESANN)*, 2013.

[31] M. Zhang and A. Sawchuk, "USC-HAD: A daily activity dataset for ubiquitous activity recognition using wearable sensors," in *ACM Int. Conf. on Ubiquitous Computing Workshop on Situation, Activity and Goal Awareness (SAGAware)*, 2012.

[32] A. Khan, N. Hammerla, S. Mellor, and T. Pltz, "Optimising sampling rates for accelerometer-based human activity recognition," *Pattern Recognition Letters*, vol. 73, pp. 33 – 40, 2016.

[33] R. Kitchin, "Big data, new epistemologies and paradigm shifts," *Big Data & Society*, vol. 1, no. 1, pp. 1–12, 2014.

[34] I. Hrynaszkiewicz, M. L. Norton, A. J. Vickers, and D. G. Altman, "Preparing raw clinical data for publication: guidance for journal editors, authors, and peer reviewers," *Trials*, vol. 11, no. 1, p. 9, 2010.