# Ultra-Low Power Context Recognition Fusing Sensor Data from an Energy-Neutral Smart Watch

Michele Magno[1,2(✉)], Lukas Cavigelli[1], Renzo Andri[1], and Luca Benini[1,2]

[1] ETH Zurich, Rämistrasse 101, 8092 Zurich, Switzerland
{magno, cavigelli, benini}@iis.ee.ethz.ch
[2] University of Bologna, Viale Risorgimento 2, 40136 Bologna, Italy

**Abstract.** Today sensors and wearable technologies are gaining popularity, with people increasingly surrounded by "smart" objects. Machine learning is used with great success in wearable devices and sensors in several real-world applications. In this paper we address the challenges of context recognition on low energy and self-sustainable wearable devices. We present an energy efficient multi-sensor context recognition system based on decision tree to classify 3 different indoor or outdoor contexts. An ultra-low power smart watch provided with a micro-power camera, microphone, accelerometer, and temperature sensors has been used to real field tests. Experimental results demonstrate both high mean accuracy of 81.5 % (up to 89 % peak) and low energy consumption (only 2.2 mJ for single classification) of the solution, and the possibility to achieve a self-sustainable system in combination with body worn energy harvesters.

**Keywords:** Ultra-low power · Smart watch · Context recognition · Machine learning · Sensor fusion · Energy neutral · Feature selection

## 1 Introduction

Today sensors and embedded technologies are gaining popularity, with people increasingly surrounded by embedded sensing devices. Smart, connected products are made possible by vast improvements in processing power and device miniaturization, and by the availability of ubiquitous wireless connectivity. Driven by Moore's Law these devices have become smaller and smaller and new applications are now possible. A fast growing class of such devices is "smart wearables", where electronics and sensors are tightly coupled with the human body [1]. The largest hi-tech companies, such as Google, Samsung, and Apple, have either already launched wearable consumer products, or are in the process of creating prototypes in an effort to fuel the next wave of exponential growth in the consumer market. Wearable technology is also very important for healthcare where electronic smart devices can continuously monitor patient health data and enable doctors to identify possible diseases earlier and to provide optimal treatment [2, 3].


B. Mandler et al. (Eds.): IoT 360° 2015, Part II, LNICST 170, pp. 331–343, 2016.
DOI: 10.1007/978-3-319-47075-7_38

The clear trend is that wearables are becoming ubiquitous in our lives replacing classically non-electric items like shoes or clothes. Soon a trillion sensor-rich connected devices are going to produce a mind-boggling quantity of data and potentially useful information [4]. However, data alone do not provide value unless we can turn them into actionable, contextualized information. Big data and data visualization techniques allow us to gain new insights through batch-processing and off-line analysis. Real-time sensor data analysis and decision-making is often done manually, but to make it scalable, it should preferably be automated.

The major challenge for wearable systems is to be able to understand the world in a similar way humans do [5]. Perceptive low-power sensor devices should be able to interpret the context around their users and allow context-aware multi-agent interaction. In fact, human activity and context recognition is the key technology for achieving pervasive computing applications from home automation to healthcare, from sport and fitness to augmented reality. Machine learning technologies are used with great success in many application areas, solving real-world problems in entertainment systems, robotics, health care, and surveillance [6–10] and are becoming essential also to wearable applications [11, 12]. For example, helping athletes by providing motion sequence analysis, or detecting abnormal situations for elderly or patient care.

More and more researchers are tackling action and classification problems with algorithms which deal with feature extractors and classifiers with lots of parameters that are optimized using the unprecedented wealth of data that has recently become available. These techniques are achieving record-breaking results, and have started outperforming humans on very challenging problems and datasets, and surpass more mature ad-hoc approaches trying to model the specific problem at hand [7–17]. However, machine learning approaches are still a challenge for low power devices such as wearables, because, in their current embodiments, they still require massive amounts of computational power. Current wearable sensor technologies do not analyze data on-board and usually leverage smart-phones for computationally intensive activity (such as sitting, standing, walking, and running) monitoring. As opposed to conventional monitoring systems that send the sensor data to a datacenter or mobile phones to be stored and processed, embedded *smart systems* process the data partially, or fully, in situ. This can significantly reduce the amount of data to be transmitted and the required human intervention – the sources of the two most expensive aspects of distributed sensing.

A typical wearable device consists of a battery-powered computing unit, a wireless communication interface, sensors, and power supply packaged in a small and unobtrusive form factor suitable to be attached to the human body [12]. Thus, low power design and software optimization is even more challenging in wearable systems due to the limited energy availability of the battery and computational resources. In fact low power design alone is not enough to make these devices with battery lifetimes of months or years, instead of just mobile, with daily battery recharges, similar to today's smart phones. Power consumption reduction with power managed resources and low power software improves and extends the lifetime of battery-operated devices. Another method for re-charging the available energy stored in batteries or super capacitors is by using energy harvesters that collect energy from the environment is the most adopted technology. Researchers have been very active in this field and energy harvesting is today a well-understood technique. However most of the presented approaches usually imply an outdoor setting, using solar panels, wind turbines or high-frequency vibration as the energy source. Solar energy in particular has been demonstrated as optimal solution for achieving self-sustainability in many outdoor applications when the entire device is carefully dimensioned and designed. In the general case, where the device cannot be assumed to be continuously operating in an outdoor environment, energy harvesting still remains challenging. Energy harvesting for mobile/wearable devices is even more challenging due to the more stringent size/weight constraints, which limits the size of harvesters and energy storage devices [15].

This work focuses on low power smartwatch devices and on context recognition in our daily lives. The main contributions of this work are as follows: design and optimization of an energy efficient smart watch to perform context recognition with low power heterogeneous sensors; using an ultra-low power camera and microphone coupled with inertial and temperature sensors to improve the classification accuracy; investigation of a low power on board feature extraction and classification with a low power and limited computation resources microcontroller (MSP430 from Texas Instruments); evaluation of the energy efficiency and self-sustainability of the context recognition on a smart watch achieving an unobtrusive monitoring system. The key feature of the system is its low power algorithm as well as the heterogeneous system implemented allowing monitoring in diverse situations. The main goal is to explore the feasibility of low power multi-sensors classification algorithms with data fusion, and the benefits of a combination of hardware and software co-design to achieve a self-sustainability when the system works with energy harvesters (i.e. solar cells). This paper is organized as follows: Sect. 2 describes the existing work on using sound to monitor beehives; Sect. 3 describes the system architecture Sect. 4 outlines the low power context recognition; Sect. 5 shows experimental results of the in-field implementation of the demonstration system; and Sect. 6 concludes the paper.

## 2 Related Work

Research on mobile and wearable sensors systems has been very prolific in recent years with a variety of solutions in a wide range of application scenarios [3, 5]. Between them, there are many examples of implemented and deployed wearable devices that attempt to exploit intelligent sensing, wireless communication and computing abilities to monitor human activities [1]. Machine learning has been applied in a wide range of applications [7, 8], and in the field of embedded sensor devices has been very active in especially due to the Smartphone's increased computational power and the availability of on-board MEMS sensors (i.e. accelerometers, gyroscope) [9, 11, 12]. In fact, many recent works use Smartphone's and MEMS for activity recognition, crowd sensing, fall detection among many others [6, 12, 20]. For this reason, there are a huge number of classification algorithms from the machine learning area for smartphones, including decision trees, k-nearest neighbors, support vector machines (SVMs), naïve Bayes and more recently neural networks (NNs) [12–14]. A more detailed analysis is presented in [15] with accelerometers on both wrists, shoulder, legs, hip, and both ankles and different combinations of them. Accuracy with only one accelerometer on the left wrist was between 5 % and 95 % for 8 classes (sitting, standing, walking, upstairs, and downstairs, handshake, whiteboard and keyboard). Better results were achieved with accelerometer on the shoulder, wrist and elbow where accuracies between 40 % and 99 % has been achieved and 85 % in average when one sensor was placed on one leg. Another interesting study was done by Porzi et al. [17], where a system was built on the Sony ecosystem of Sony Xperia Z smartphone and Sony SmartWatch. The system was implemented using a smartphone for gesture recognition for use by the visually impaired. They presented an optimized kernel method (global alignment kernel) for discrete-time warping in SVMs. Discrete time warping allows to map similar gestures when moving at different speeds. Secondly, they implemented logo recognition like the "wet floor sign" where the camera of the smartphone was used. A recognition rate of 95.8 % was achieved.

Our works focuses on machine learning optimized for low power microcontrollers with limited resources, and on using sensor fusion, with camera and microphone, instead of a single low power sensor, such as a motion sensor. In the proposed work we investigate algorithms which can process the data close to the sensors instead of sending the data to a remote host or smartphone. Recently, many approaches tried to classify users' activities by deploying several heterogeneous sensors on the human body such as accelerometers, camera, acoustic, and temperature to capture characteristic repetitive motions, postures, and sounds of activities [16, 18]. However, the main challenges of wearable design are to prolong the operating lifetime and to enhance usability, maintenance, and mobility, while keeping a small and unobtrusive form factor. Low power embedded machine learning is still challenging due to the limited computational resources, limited power budget and the high requirements of the algorithms [21]. In this work we focus in low power heterogeneous sensors, optimizing the hardware and energy-efficient high accuracy machine learning algorithms to achieve a self-sustainable system.

## 3 System Architecture

Figure 1 shows the architectural overview of the smart watch whit different voltage domains and power switches to achieve energy efficiency. The core of the proposed hardware consists of the microcontroller TI MSP430FR5969. This 16-bit microcontroller has 2 kB of SRAM and 64 kB of non-volatile FRAM which uses less power than other non-volatile memories and reduces the gap in speed with the SRAM. The MSP430 also supports different low power modes which allows us to decide which components of the microcontroller are supplied. In low power mode LP4.5 it is praised to typically use 20 nA and in active mode 800 µA at a clock frequency of 8 MHz. The proposed system is equipped with a camera, a microphone, an accelerometer, and a temperature sensor. Furthermore the device can communicate using NFC and optionally by Bluetooth when the layer 2 board of the smart watch containing a Bluegiga WT12 module is attached. As an alternative, an external device can be attached to the UART pins by the main pin socket. Figure 2 shows the developed smart watch used to collect sensor data in real life indoor and outdoor scenarios. Additional memory is provided by a micro SD card holder with which a micro SD card can be

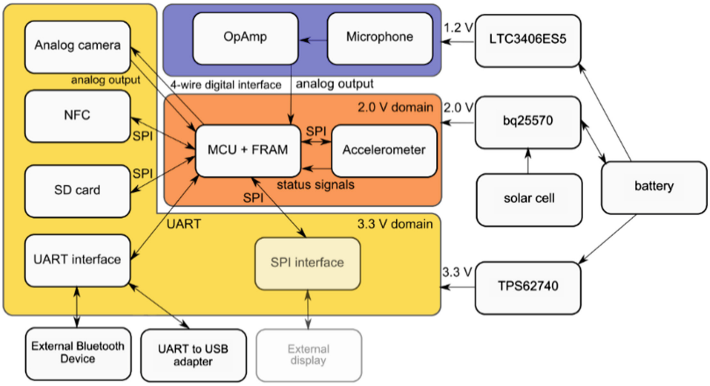


**Fig. 1.** Overview of the system architecture.

connected to the microcontroller using SPI. To minimize possible power consumption, each sensor can be completely switched off by a low-leakage load-switch with a quiescent current of 240 nA. The temperature sensor is directly supplied by an output pin from the microcontroller such that power is only consumed when temperature is measured and no additional load switch is needed. A load-switch for the accelerometer was resigned because leakage current in standby mode is only 10 nA.

**Energy Harvesting and Voltage Suppliers.** The device supports a power harvester chip TI BQ25570, which manages a LiPo rechargeable battery and solar cells. The internal DC-DC converter of the harvester chip can be set to a variable output voltage. This has been used to realize voltage scaling from 3.0 V down to 2.0 V, leading to much lower power consumption because dynamic power is related to the voltage squared:

$$P_{dyn} = \alpha C V^2 f,$$

where $\alpha$ is the switching rate, $C$ is the load capacitance, $V$ is the supply voltage and $f$ is the operating frequency. The power harvester contains a highly efficient boost converter and supports maximum power point tracking.

In order to optimize the power consumption there are three different power domains. The peripherals are supplied with the buck converter TPS62740 from Texas Instruments which has an operating quiescent current of 460 nA and typically 70 nA in shutdown mode. The microphone is supplied separately with 1.2 V by the buck converter LTC3406ES5-1.2.

**Camera.** The camera is the ultra-low power $112 \times 112$ pixel gray-scale CMOS camera Centeye Stonyman, which has a focal plane size of 2.8 mm × 2.8 mm and a pixel pitch of 25 μm [22] that consumes only 2 mW@3.3 V. The camera comes on a pre-soldered PCB containing the image sensor and an objective lens and is connected to the smart watch by a socket connector. The camera provides an analog output which is connected with the internal ADC of the MSP430 [19].

**Accelerometer.** The accelerometer used is a ULP ADXL362 from Analog Devices with high resolution down to $9.8 \cdot 10^{-3}$m/s$^2$. It needs 1.8 μA while sensing at 100 Hz and only 10 nA in standby mode and provides a burst mode including a FIFO buffer. This mode allows consecutive reading of the acquired sensor data. The microcontroller can do other jobs in parallel or enter a low-power mode. The accelerometer is connected to the microcontroller via the SPI interface and with two status signals. These status signals can be used to interrupt or wake the microcontroller up, when a predefined event happens like acceleration exceeds some threshold or the FIFO-buffer is full.

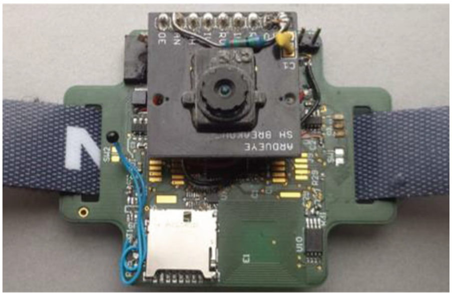

**Fig. 2.** Picture of the smart watch.

**Microphone.** The microphone which was used is the low-power microphone INMP801 which was mainly designed for hearing aids and consumes 17 μA at a supply voltage of 1.2 V and outputs a voltage of 570 ± 159 mV. The audio signal is amplified by a TI LMV951. Also this sensor is connected to the internal ADC of the MSP430.

**Temperature Sensor.** The temperature sensor is a Negative Temperature Coefficient Thermistor (NTC) thermistor from Epcos/TDK which is used in a voltage divider configuration and connected to the ADC.

## 4 Low-Power Context Recognition

Context awareness provides completely new use cases for a smart watch and makes it much more user-friendly. Figure 3 shows the software stack of the smart watch, where the close interaction between hardware and software directly on board can be seen. Our context recognition tries to classify the context of the action that is being performed by the wearer and the surroundings of the smart watch. It does so based on the data available from the many different sensors.

In order to train a classifier we need to collect and label a dataset. We chose 3 classes: public transport, office, and cafeteria, and acquired data from the temperature sensor, accelerometer, camera and microphone. Each data item lasts 5 s and contains 8 kHz audio data, 100 Hz 3-axis accelerometer measurements, one temperature readout and one image of 112 × 112 grayscale pixel. Each of the classes has several hours of labeled data.

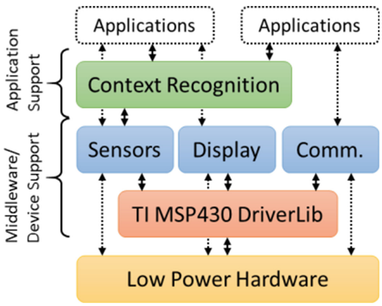


**Fig. 3.** Software stack of the smart watch

### 4.1 Feature Extraction and Selection

Performing classification on the acquired raw sensors data directly yields very poor results, because they usually represent the information in an unfavorable way, e.g. such that very little noise or small variations of the environment yield orthogonal representations. This is overcome by extracting features from this data. For the different types of sensors, there are different suitable features. For the sensors hosted by developed smart watch the possibility are as follow:

- For the audio data, we use the number of zero crossing with a 1 % hysteresis, the average energy of the signal, the maximum absolute value and dispersion. We also use features from the frequency domain, such as the spectral centroid, the bandwidth, and the well-known Mel-frequency cepstral coefficients (MFCCs).
- For the accelerometer data, we compute mean, variance, energy, covariance between the axes, the dynamic range and the frequency-domain entropy.
- For the temperature we calculate the average rate of change, mean, variance, and dynamic range.
- For the camera data we computed the mean (avg. brightness), variance and contrast.

Many of these features have an intuitive meaning, like the energy of a segment of the audio stream, which provides an indication of the loudness. In total we found 65 different features which are summarize in Table 1. There are 27 time domain feature, (4 for microphone, 16 for the accelerometer, 4 for the temperature sensor, 3 for the camera) and 38 features in frequency domain for microphone and accelerometer. Among the microphone feature there are 14 using Discrete Furier Transform (DFT) and 14 using Mel frequency.

**Table 1.** Total pool of features for different sensors.

| Sensor | Time | Frequency | Total |
|---|---|---|---|
| Microphone | 4 | 4 + 14(DFT) + 14(Mel) | 36 |
| Accelerometer | 16 | 6 | 21 |
| Temperature | 4 | | 4 |
| Camera | 3 | | 3 |
| Total | 27 | 38 | 65 |

However, a large amount of features will increase the evaluation energy, and as our first goal is to have an energy-efficient system, we chose to perform feature selection, keeping only a fixed number of features.

Ideally we would choose a feature set $\hat{S} \subseteq \bar{S}$ of some fixed cardinality $\left|\hat{\mathcal{S}}\right| = N$ among all features $\bar{\mathcal{S}} = \bigcup_i \mathcal{F}_i$, such that we maximize the mutual information between the selected features and the set of target classes $\mathcal{C}$, i.e.

$$\hat{\mathcal{S}} = \mathrm{argmax}_{\mathcal{S}} D(\mathcal{S}, \mathcal{C}), \qquad D(\mathcal{S}, \mathcal{C}) = I(F_1, \ldots, F_N; C).$$

Solving this optimization problem is called *max-dependency* feature selection. However, with limited training data the estimated densities $p(x_1, \ldots, x_N)$ and $p(x_1, \ldots, x_N, c)$, where $x_1, \ldots, x_N$ are possible values for the various features and $c \in \mathcal{C}$ is the target class, are not very accurate. This makes it pointless to solve the above optimization problem, since the mutual information cannot be calculated with reasonable precision.

A simple approximation of the above problem is looking for the most relevant features, maximizing the mutual information individually with $D(\mathcal{S}, \mathcal{C}) = \frac{1}{|\mathcal{S}|} \sum_{i=1}^{N} I(F_i; C)$. Among the best features according to the solution of the maximum relevance problem, these are the mean of the temperature, the mean of the camera image, the spectral energy and entropy of the accelerometer axes, followed by a long list of audio features.

### 4.2 Classification

**Input** : Labeled training data $\left\{\left(x^{(i)}, \ell^{(i)}\right)\right\}_{i \in \{1,2,\ldots,m\}}$, $\ell^{(i)} \in \mathcal{C}$
**Input** : Maximum tree depth $d_{max}$ and minimum number of items per node $m_{min}$
**Output**: Decision tree $T$
C45_MakeTree($\{x^{(i)}, \ell^{(i)}\}_i, d = 1$) **begin**

```
    /* Stop descent if all samples are correctly classified, there are too few
       elements left, or we have reached that maximum depth:               */
    if ℓ^(i) = ℓ^(j) ∀i,j or m < m_min or d > d_max then
        return leaf of class ĉ = arg max_c p_c with p_c = (1/m) Σ_i 1{ℓ^(i) = c}
    end
    /* Calculate entropy of current node:                                   */
    H_root ← − Σ_{c∈C} p_c log2(p_c)
    /* For each feature k find the threshold t̂_k maximizing inf. gain:     */
    foreach feature k do
        t̂_k ← arg max_{t_k} (H_root − p_(x_k≤t_k) H_(x_k≤t_k) − p_(x_k>t_k) H_(x_k>t_k))
    end
    /* Find feature with highest information gain:                          */
    k̂ ← arg max_k (H_root − p_(x_k≤t̂_k) H_(x_k≤t̂_k) − p_(x_k>t̂_k) H_(x_k>t̂_k))
    /* Create two subtrees T1 and T2:                                       */
    T_left ← C45_MakeTree({x^(i), ℓ^(i)}_{x^(i)≤t̂}, d + 1)
    T_right ← C45_MakeTree({x^(i), ℓ^(i)}_{x^(i)>t̂}, d + 1)
    return (k̂, t̂_k̂, T_left, T_right)
end
```

**Algorithm 1:** Continuous C4.5 Algorithm

There are many well-known classification algorithms and concepts, ranging from simple decision trees to support-vector machines, nearest-neighbor algorithms and boosting to random forests, neural networks, naïve Bayes and complex graphical models, to name just a few. We perform the classification using a decision tree which is constructed using the continuous C4.5 algorithm (cf. Algorithm 1). The individual features of a data sample $x$ are addressed by subscript, i.e. $x_1$ denotes the first feature. The decision tree was chosen because of its low computational complexity and consequential high energy efficiency during classification (as opposed to during learning) [1]. For performing the classification, there is only the decision tree, which has to be descended doing the corresponding comparisons until arriving at a leaf.

Decision trees are very susceptible to overfitting, so particularly with our continuous-valued and limited amount of training data this is an issue. It can be approached using bottom-up pruning, where leaves with only few samples (below some threshold) are combined to a single leaf of the most probable class. Pruning is also done offline and can only improve the time required for classification.

## 5 Measurement Results

In a first step, we tune the feature selection to choose the optimal number of features. The feature selection and construction of the decision tree were performed on the training set. The training set has been acquired using 7 subjects wearing the smart watch for 24 h and collecting data from all sensors. For each class a minimal number of 500 samples have been used for the training set. The results shown in Fig. 4 are the classification accuracy of the test set. The mean accuracy was best when selecting the 21 features of maximal relevance and minimal redundancy. With the low-complexity classification system presented here, we were able to achieve a mean accuracy of 81.5 % for all classes using 21 features.

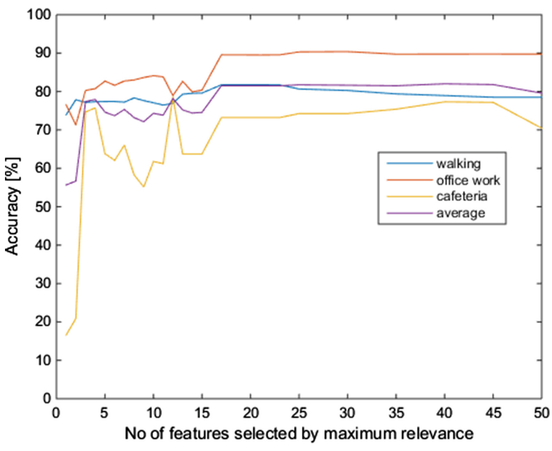


**Fig. 4.** Class-wise 1-vs-all accuracy depending on the number of features selected using the relevance evaluated with mutual information with data from all sensors.

| Ø 81.5% | | PT | Office | Cafetteria. |
|---|---|---|---|---|
| | Publ. Transp. | **81.8** | 13.6 | 4.6 |
| | Office | 7.2 | **89.5** | 3.3 |
| | Cafeteria | 17.2 | 9.6 | **73.2** |

**Fig. 5.** Confusion Matrix using 21 features.

To give more insight into the limitations, we present the confusion matrix in Fig. 5. The classes are very well identified from 73.2 % to 89.5 %, and we are expected to achieve even better performance when training the system with more subjects and data.

**Energy vs. Accuracy.** An important evaluation we measured is the link between energy used by the combination of sensors and the accuracy achieved. As the various sensors require a substantial amount of power, clearly, there is a trade-off between which sensors are used and what accuracy can be achieved. We visualize this trade-off in Fig. 6, considering only the data of some of the sensors mentioned. The energies presented here are based on the measurements during sensor data acquisition and estimates based on counts of the number of required operations for the feature extraction. As expected, utilizing all sensors achieves the best accuracy but also has high energy consumption. Another interesting result is the improved performance (in energy and accuracy) of the ultra-low power camera against the accelerometer which commonly is considered a low power high accuracy sensor. This demonstrates that the accelerometer is an ideal sensor for detecting motions (i.e. walking, running, gesture, etc.) but can be overcome in the context detection. In fact the camera needs less than 100 ms (so very small energy required) to acquire a frame but gives a lot of information while the accelerometer needs seconds (at least 1 in our classificatory) to acquire sufficient data for classification. Also it is interesting that the microphone over performs

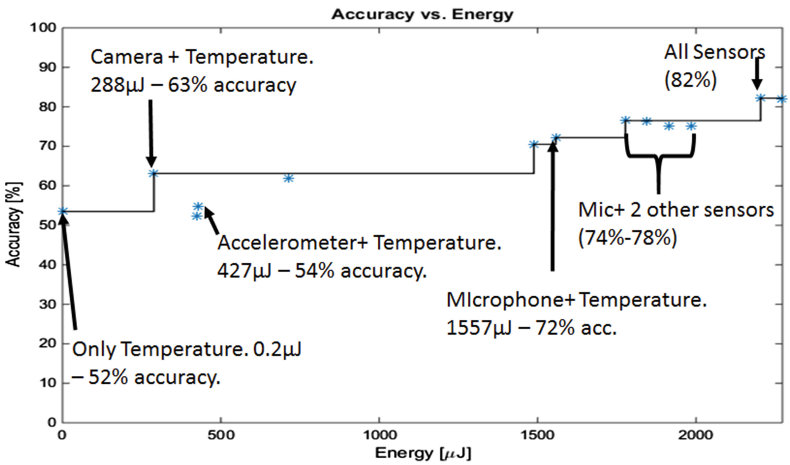


**Fig. 6.** Achieved accuracy by used sensors with annotated pareto-optimal points.

| Activity | Run time [d] |
|---|---|
| Idle | 661.38 |
| Every day | 660.92 |
| Every hour | 650.69 |
| Every 30 minutes | 640.34 |
| Every minute | 333.10 |
| Every 10 seconds | 95.67 |
| Permanent | 11.00 |

| Activity | Run time [d] |
|---|---|
| Every 745 sec. | |
| Every 5 minutes | 5617.52 |
| Every minute | 729.89 |
| Every 10 seconds | 113.37 |
| Permanent | 11.20 |

**Fig. 7.** Life time and self-sustainability analysis. This figure shows the life time based on one full battery charge without and with energy harvesting in the left and right table, respectively.

both the camera and accelerometer in accuracy but, of course, it is more expensive in terms of energy, as more energy is required by the sensor and frequency domain feature computation.

**Lifetime Estimation.** To evaluate the life time and self-sustainability of the proposed solution, we measured the energy for the acquisition of all sensors and feature calculation. The evaluation setup was preformed acquiring 1 picture for the camera, 1 data from the temperature sensor, 1 s of accelerometer data, and 1 s of microphone data, and we compute all 65 features for the classification algorithm to have a worst case. With this set up the energy needed for single classification is only 2.28 mJ. To evaluate the lifetime we assume that we use a small Li-Ion battery with 150 mAh capacity and 4.2 V. For the power harvester which we used an average value of the calculations are based on an average power generation of 40 μW. As we demonstrated in previous work [19] this is a pessimistic value that can be easily harvested from a wrist band with eight $1 \times 4$ cm solar cells. To evaluate the idle energy, when the smartwatch is not performing any classification, we measured the quiescent power of the developed version of smart watch (only 9 μW). Figure 7 shows the different lifetime according to the classification duty cycle. It can be observed that when there is no acquisition and no harvesting, the device can last for more than 661 days, which highlights the low quiescent energy. When the features are calculated continuously for all sensors, 11 days are possible. If a sensor acquisition is taken every 10 s, the device could last for more than 95 days. If this is further reduced to a periodicity of once per day, 660 days are possible. When considering the power harvester is plugged in, the device can do classifications every 10 s for more than 113 days. Self-sustainability is reached when a classification is performed every 745 s. But with a 5-min cycle more than 5,617.52 days are possible (15.5 years).

## 6 Conclusions and Future Work

We implemented a prototype of smart watch trained to recognize context directly on board and achieve self-sustainability. The smartwatch hosts an ultra-low power gray-scale camera, a MEMS microphone, a 3-axes accelerometer, and an analog temperature sensor. Moreover the device is equipped with a solar harvester and rechargeable Li-Ion battery to continuously recharge the battery even in indoor scenario. The implemented classifier is suitable for ultra-low power microcontrollers and this system demonstrates the recognition of various scenarios directly on board the existing smart watch, an accelerometer alone cannot achieve this without a smart phone. The experimental results confirmed both the benefits of the data fusion and the energy efficiency of the solution. Even if the dataset used for the training was not huge the preliminary results show a mean accuracy of 82.5 % when classifying 3 cases with peak of 89 %. In future work we are planning to improve the training dataset to have more labelled classes and that should increase the overall accuracy of the algorithm. Moreover, a full version of the context recognition algorithm will be implemented on the smart watch and tested in the field.

**Acknowledgments.** This work was supported by the SCOPES SNF project (IZ74Z0_160481).

## References

1. Ghasemzadeh, H., Jafari, R.: Ultra low-power signal processing in wearable monitoring systems: a tiered screening architecture with optimal bit resolution. ACM Trans. Embed. Comput. Syst. (TECS) **13**, 9 (2013)
2. Kahn, J., Yuce, M.R., Bulger, G., Harding, B.: Wireless Body Area Network (WBAN) design techniques and performance evaluation. J. Med. Syst. **36**(3), 1441–1457 (2012)
3. Hung, K., et al.: Ubiquitous health monitoring: integration of wearable sensors, novel sensing techniques, and body sensor networks. In: Adibi, S. (ed.) Mobile Health, pp. 319–342. Springer, Heidelberg (2015)
4. Pejovic, V., Musolesi, M.: Anticipatory mobile computing: a survey of the state of the art and research challenges. ACM Comput. Surv. (CSUR) **47**(3), 47 (2015)
5. Perera, C., et al.: Context aware computing for the Internet of Things: a survey. IEEE Commun. Surv. Tutor. **16**(1), 414–454 (2014)
6. Anjum, A., Ilyas, M.U.: Activity recognition using smartphone sensors. In: IEEE Consumer Communications and Networking Conference (CCNC) (2013)
7. Sharma, S., et al.: Machine learning techniques for data mining: a survey. In: IEEE International Conference on Computational Intelligence and Computing Research (ICCIC) (2013)
8. Govindaraju, V., Rao, C.: Machine Learning: Theory and Applications. Elsevier, New York (2013)
9. Azizyan, M., et al.: SurroundSense: mobile phone localization via ambience fingerprinting. In: Mobicom, pp. 261–272 (2009)
10. Kerhet, A., et al.: Distributed video surveillance using hardware-friendly sparse large margin classifiers. In: IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 87–92 (2007)
11. Chon, Y., et al.: Mobility prediction-based smartphone energy optimization for everyday location monitoring. In: SenSys 2011, pp. 82–95 (2011)
12. Lara, O.D., Labrador, M.A.: A survey on human activity recognition using wearable sensors. IEEE Commun. Surv. Tutor. **15**(3), 1192–1209 (2013)
13. Gokgoz, E., Subasi, A.: Comparison of decision tree algorithms for EMG signal classification using DWT. Biomed. Signal Process. Control **18**, 138–144 (2015)

14. Silva, J., et al.: Human activity classification with inertial sensors. In: PHealth: International Conference on Wearable Micro and Nano Technologies for Personalized Health, vol. 200, p. 101 (2014)
15. Weddell, A.S., et al.: A survey of multi-source energy harvesting systems. In: Conference on Design, Automation and Test in Europe (DATE), pp. 905–908 (2013)
16. Maurer, U., et al.: eWatch: a wearable sensor and notification platform. In: International Workshop on Wearable and Implantable Body Sensor Networks (BSN), p. 145 (2006)
17. Porzi, L., et al.: A smart watch-based gesture recognition system for assisting people with visual impairments. In: ACM International Workshop on Interactive Multimedia on Mobile & Portable Devices (IMMPD), pp. 19–24 (2013)
18. Maekawa, T., Yanagisawa, Y., Kishino, Y., Ishiguro, K., Kamei, K., Sakurai, Y., Okadome, T.: Object-based activity recognition with heterogeneous sensors on wrist. In: Kay, J., Lukowicz, P., Tokuda, H., Olivier, P., Krüger, A. (eds.) Pervasive 2012. LNCS, vol. 7319, pp. 246–264. Springer, Heidelberg (2010). doi:10.1007/978-3-642-12654-3_15
19. Magno, M., et al.: InfiniTime: a multi-sensor energy neutral wearable bracelet. In: International Green Computing Conference (IGCC) (2014)
20. Seong, K.E., et al.: Self M2M based wearable watch platform for collecting personal activity in real-time. In: IEEE Conference on Big Data and Smart Computing (BIGCOMP), pp. 286–290 (2014)
21. Zhu, Z., et al.: Fusing on-body sensing with local and temporal cues for daily activity recognition. In: ICST International Conference on Body Area Networks, pp. 83–89 (2014)
22. Centeye, Inc., Centeye Stonyman/Haskbill silicon documentation (2013)