Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 207 (2022) 3415–3423

Procedia Computer Science

www.elsevier.com/locate/procedia

26th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems (KES 2022)

Embedded AM-FM Signal Decomposition Algorithm for Continuous Human Activity Monitoring

Giorgio Biagetti[a], Paolo Crippa[a,*], Dario Bocchini[b], Michele Alessandrini[a], Laura Falaschetti[a], Claudio Turchetti[a]

[a]*DII - Department of Information Engineering, Università Politecnica delle Marche, via Brecce Bianche, 12, I-60131 Ancona, Italy*

[b]*Università Politecnica delle Marche, via Brecce Bianche, 12, I-60131 Ancona, Italy*

Abstract

AM-FM decomposition techniques have been successfully used for extracting significative features from a large variety of signals, helping realtime signal monitoring and pattern recognition, since they represent signals as a simultaneous composition of amplitude modulation and frequency modulation, where the carriers, amplitude envelopes, and the instantaneous frequencies are the features to be estimated. Human activities often involve repetitive movements, such as in running or cycling, where sinusoidal AM-FM decompositions of signals have already demonstrated to be useful to extract compact features to aid monitoring, classification, or detection. In this work we thus present the challenges and results of implementing the iterated coherent Hilbert decomposition (ICHD), a particularly effective algorithm to obtain an AM-FM decomposition, within a resource-constrained and low-power ARM Cortex-M4 microcontroller that is present in a wearable sensor we developed. We apply ICHD to the gyroscope data acquired from an inertial measurement unit (IMU) that is present in the sensor. Optimizing the implementation allowed us to achieve real-time performance using less then 16 % of the available CPU time, while consuming only about 5.4 mW of power, which results in a run-time of over 7 days using a small 250 mAh rechargeable cell.

Keywords: AM-FM decomposition; human activity monitoring; human activity recognition (HAR); feature extraction; iterated coherent Hilbert decomposition (ICHD); Inertial monitoring unit (IMU); microcontroller; embedded system; wearable sensor

1. Introduction

In the last decade classifying, recognizing or simply monitoring human activity all benefitted of the availability of wearable tiny body sensors. These tasks have now become easy-to-be-implemented, and useful for improving sports,

* Corresponding author. Tel.: +39-071-220-4541; fax: +39-071-220-4464.
E-mail address: p.crippa@univpm.it

1877-0509
10.1016/j.procs.2022.09.401

fitness, healthcare, ambient assisted living and telemedicine research fields [1–4]. Indeed, acquiring heterogeneous data such as body position and movement, heart rate, muscle fatigue, and skin temperature, simply using smartwatches or clothes, belts, shoes, sunglasses, in which miniaturized and lightweight sensors have been transparently embedded, is quite common nowadays [5–8]. Among all these heterogeneous data, body accelerations are the most widespreadly employed to evaluate human activities as they record body movements and detect variations in limb orientation [9–12]. Thus, they have been exploited by many wearable-sensor-based human activity recognition (WSHAR) systems [13–17].

Accelerometer sensors placed across the body often deal with periodic non-stationary signals during recreational, sports, training and rehabilitation activities involving repetitive movements such as walking, running, cycling, weightlifting. On this scenario natively non-stationary signal processing algorithms can be useful to estimate the time-dependent spectral moments. In particular, the empirical mode decomposition (EMD) and its modifications [18], have been applied to the processing of biological data [19, 20].

For repetitive movements such as running or cycling, sinusoidal AM-FM decompositions of signals have demonstrated to be useful because they represent signals themselves as resulting from simultaneous amplitude modulation and frequency modulation, where the carriers, amplitude envelopes, and instantaneous frequencies (IFs) are the features to be estimated. Monocomponent representations are consolidated models and several demodulation algorithms exist, as those based on the Teager-Kaiser operator or the Hilbert transform [21–23]. These approaches use *ad hoc* filters in order to regularize the estimated envelope and IF of the signal. Multicomponent AM-FM representations, that are better suited to model nonlinear and non stationary signals, have been applied in a large number of research fields [24–33].

In general AM-FM decomposition techniques have been successfully used for extracting significative features from a large variety of signals, helping realtime signal monitoring and pattern recognition, e.g. in [34] a multiscale AM-FM decomposition has been converted from MATLAB code into compute unified device architecture (CUDA) code, in order to take advantage of the graphics processing units to significantly reduce the computation time and memory needs, while in [35] an efficient FPGA implementation of EMD is presented. But neither CUDA nor FPGA are particularly suited for wearable devices that must run for prolonged period of times on small and lightweight batteries, and implementations of these types of decomposition in low-power microcontrollers are not common.

In this work we thus present the challenges and results of implementing the iterated coherent Hilbert decomposition (ICHD) [7], a particularly effective algorithm to obtain an AM-FM decomposition, within the resource-constrained and low-power microcontroller that is present in the wearable sensor [2] we developed. It was designed specifically to monitor activities of daily living of people with cognitive impairments, so it must have a long battery life to allow continuous operation, but of course it can be used also to monitor and track fitness and recreational activities. This device incorporates an inertial platform consisting in a 3-axes linear accelerometer and a 3-axes gyroscope, able to sense angular speed. It is able to stream the acquired data over a Bluetooth low energy (BLE) wireless channel, using an embedded low-power transceiver, to any suitable receiving device, such as a smartphone or a laptop. ICHD was already proved to be able to extract very compact and meaningful features for the classification of human activities [36], so the possibility of using it aboard the device could provide significant bandwidth savings by only transmitting these compact features instead of the whole motion signal.

2. Methods

2.1. Iterated Coherent Hilbert Decomposition

The details of ICHD are thoroughly described in [7], this section reports a brief summary for easier reference.

ICHD is based on the extraction of quasi-sinusoidal components from a time-domain signal $x(t)$, starting with the highest amplitude one, and iteratively progressing to the smaller ones. The most prominent features of the signal should hence be captured in the first few components. Usually, just two suffice for most applications.

In general, any signal $x(t)$ can be written as:

$$x(t) \simeq \sum_{i=1}^{N} a_i(t) \cos\left(\varphi_i + \int_0^t \omega_i(\tau)\,\mathrm{d}\tau\right) \tag{1}$$

where the approximation can also be an exact identity if no constraints are posed on the signals $a_i(t)$, which should be instantaneous amplitudes, and $\omega_i(t)$, which should be instantaneous frequencies. Since, to be meaningful, we require that these signals $a_i(t)$ and $\omega_i(t)$ be slowly time-varying with respect to the dynamics of the original signal, an initial phase φ_i is added to allow smoothing of $a_i(t)$ near time zero, and up to N components are used to model progressively finer details, with the assumption that, on average, $a_i > a_{i+1}$. Of course, the decomposition is not unique, but depends on the smoothness requirements imposed.

A possible and effective extraction technique starts with an exact representation of (1) with $N = 1$, obtained by means of the Hilbert transform $\mathcal{H}[\cdot]$, which can be used to produce an analytical signal $z(t)$ such that

$$z(t) = x(t) + j\mathcal{H}[x(t)] = a(t)\,\mathrm{e}^{j\varphi(t)} \tag{2}$$

which can be recast in the form (1) by posing $N = 1$, $a_1(t) = a(t)$, $\omega_1(t) = \mathrm{d}\varphi(t)/\mathrm{d}t$, $\varphi_1 = \varphi(0)$. In general, the derivative of $\varphi(t)$ as obtained by the Hilbert transform would not be smooth, so this exact decomposition is not often useful for real-world signals. We proceed by obtaining a smooth frequency estimate by low-pass filtering the initial estimate $\omega(t) = \mathrm{d}\varphi(t)/\mathrm{d}t$, let us call $\hat{\omega}(t)$ this low-pass filtered version of $\omega(t)$. After adjusting the initial phase $\overline{\delta}(0)$, the amplitude component can be estimated by coherent demodulation:

$$c(t) = z(t)\,\exp\left(-j\,\overline{\delta}(0) - j\int_0^t \hat{\omega}(\tau)\,\mathrm{d}\tau\right) \tag{3}$$

and the AM part can be posed equal to its low-pass filtered amplitude $\overline{c}(t)$, hence $\hat{a}(t) = |\overline{c}(t)|$.

Having obtained the estimates $\hat{a}(t)$ and $\hat{\omega}(t)$, they can be assigned to a component $a_i(t)$ and $\omega_i(t)$, the resulting signal subtracted from the original, and the procedure repeated for the subsequent components.

This algorithm allows the computation of instantaneous amplitudes and frequencies, that results in an expansion of the amount of data since a single time-domain signal is now represented by $2\,N$ signals. But since these signals are now slowly varying, they can be drastically subsampled. Indeed, it has been proved that taking just their median value in a window that spans a few seconds is enough to classify activities [36]. So, after computing ICHD, only the medians $\overline{A_i}$ of $a_i(t)$ and $\overline{f_i}$ of $\omega_i(t)$ from (1) are computed (in a suitably long window) and retained.

2.2. Implementation

A straightforward implementation of the algorithm just exposed might require significant computational resources, as it requires to compute Hilbert transforms, compute inverse trigonometric functions to estimate instantaneous phase, low-pass filtering, coherent demodulation, and further low-pass filtering.

Significant savings can be obtained by performing all the filtering in the frequency domain, discontinuities in inverse trigonometric functions that can lead to problems in computing the derivative can be avoided by noting that, by taking the logarithm of $z(t)$, the instantaneous phase is just its imaginary part, and the derivative of a logarithm can be computed without requiring any transcendental function evaluation since $\mathrm{d}\varphi(t)/\mathrm{d}t = \mathrm{d}\left(\Im \log z(t)\right)/\mathrm{d}t = \Im\left(1/z(t)\cdot \mathrm{d}z(t)/\mathrm{d}t\right)$, that only requires basic (complex) arithmetic. A schematic representation of the complete data processing that is implemented in the microcontroller is shown in Figure 1.

These optimizations have first been prototyped in MATLAB to ensure correct operation, and then converted to C code, suitable to run on the embedded 32 bit ARM Cortex-M4 that powers the wearable sensor, with the help of the MATLAB Coder. Indeed, the coder was not able to directly produce C code suitable to be embedded in the device firmware. Functions such as "fftfilt" could not be converted automatically due to failure in propagation of constant values. So, several hand-tuned changes had to be implemented to optimize memory usage and allocation, avoiding the placement of large vectors on the stack (which is quite small on this platform), doing without dynamic memory allocation (to avoid heap fragmentation), and to propagate constants that can be computed at compile-time instead of run-time.

In particular, by fixing the filter length to $P = 1025$, the window length to $L = 1024$, the cut-off frequency to 40 % of the Nyquist frequency, it is possible not only to precompute the filter coefficients, which is the customary thing to do, but also to specialize the overlap-and-add algorithm to the required number of blocks, 2 in this case, using 2048-points FFTs. Moreover, since to limit the effects of artifacts near block boundaries, the ICHD algorithm implemented a mirror (even) extension of the signal, that was embedded in the filtering function, that is now composed of the following steps:

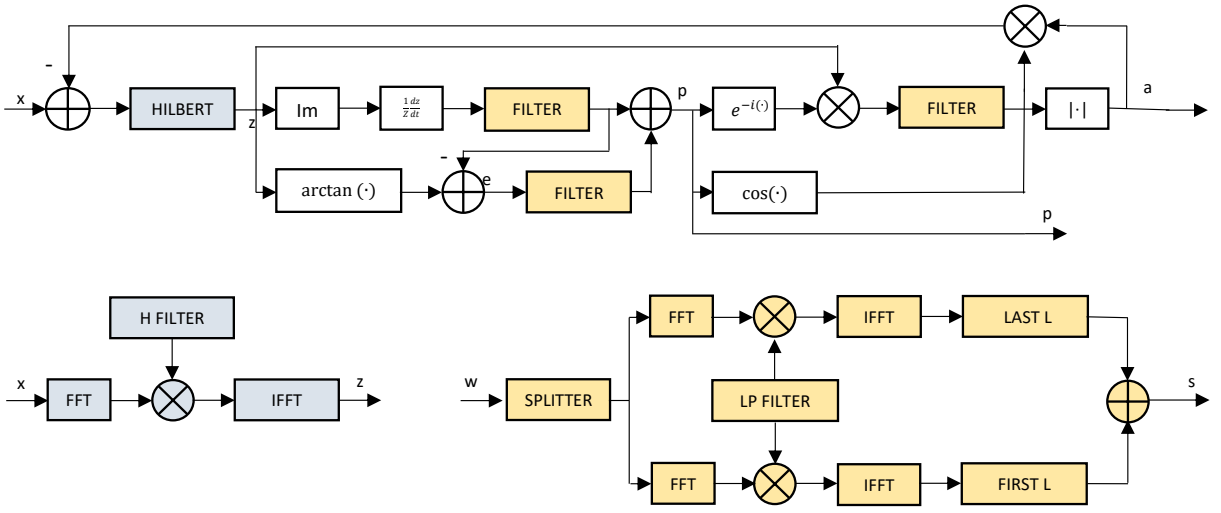

Fig. 1. Schematic representation of the data flow and of the main blocks that compose the ICHD algorithm as implemented within the microcontroller, highlighting the FFT-based implementation of the filters (Hilbert filter in gray, low-pass filter in yellow).

- preliminary step: convert the 1025-tap filter coefficients with a 2048-points FFT;
- prepare time-domain data using the first half of the acquired data, even-mirrored to the left to compensate the filter group delay.
- compute the signal FFT using the just created 1024-point data vector, zero-padded to 2048 points;
- compute the inverse FFT (IFFT) of the product of the just computed FFT with the filter frequency response;
- apply another 2048-points FFT to the second half of the acquired data, even-mirrored to the right;
- compute the inverse FFT of the product of the just computed FFT with the filter frequency response;
- add the last 1024 values from the first IFFT to the first 1024 values of the second IFFT.

With these optimizations at least a four-fold speedup was obtained w.r.t. a straightforward implementation, and further optimizations were achieved by leveraging libraries specifically optimized for the Cortex-M4 instead of relying on generic C code. For instance, using the ARM CMSIS FFT library instead of a generic radix-2 implementation more than halved again the execution time.

3. Results

To evaluate the suitability of the ICHD algorithm for real-time feature extraction in a continuous monitoring application, the C code optimized as described above was included in the firmware of the device presented in [2], called EMGyro2. It includes an STmicroelectronics LSM6DSO 6-degrees of freedom inertial sensor (gyroscope and linear accelerometer), a 6-channel EMG sensor (not used in the present work, but potentially useful for future applications always in the field of continuous human monitoring), all coordinated by a Nordic Semiconductor nRF52840 system, that incorporates the BLE radio and a 64 MHz Cortex-M4 CPU with FPU, 1 MiB of flash memory, and 256 KiB of RAM. The whole device is battery powered, and is shown in Figure 2.

A new BLE characteristic was added to the GYRO service to stream the median amplitudes and frequencies computed by the microcontroller. In the current implementation, N was fixed to 2, and the window length L to 1024. Using an acquisition frequency F_S=104 Hz, this results in windows of about 10 s, an appropriate length for activity classification. The communication requirements of the new characteristic are thus totally negligible: just four 32-bit floating point numbers every 10 s. In contrast, raw inertial data streaming for off-line elaboration requires at least 16 bits per axis, for a total of 12 bytes nearly every 10 ms, a considerable bandwidth for a BLE system that maxes out at 6×20 bytes packets every 7.5 ms.

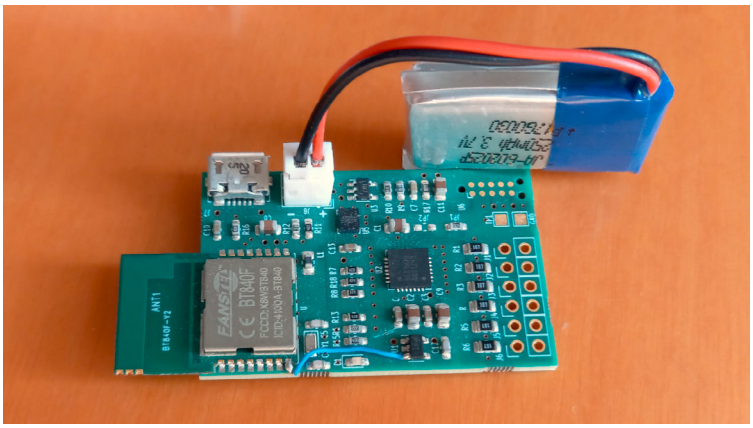

Fig. 2. Picture of the wearable device used for the experiments. PCB dimensions are 52 mm × 27 mm and it is powered by a 250 mAh lithium-polymer battery rechargeable through the micro-USB connector. The Fanstel module on the lower left contains the nRF52840 chip.

A first test to evaluate the performance of the implementation was performed by measuring the device current consumption during operation. This type of test is able to shed light on many of the internal BLE protocol operations and, most importantly, allows the precise evaluation of the CPU time required to process the algorithm and the amount of energy it consumes, hence enabling a precise estimation of the battery run time. The measurements were performed with an ad-hoc system described in [37], and the results are reported in Figure 3.

Recording of the current trace started when the device was already connected via BLE to a smartphone. After about 11.4 s the gyroscope was activated and the streaming of the raw inertial data started. The current signal seems noisy but it is actually composed of a train of pulses corresponding to radio reception/transmission of BLE packets, as better highlighted in the insets. The average current still stays quite low, at around 1.4 mA, because the packets are short. Indeed, the baseline current consumption of the gyroscope plus microcontroller, with just the keep-alive packets and no data being streamed at all, is about 1.1 mA.

After further approximatively 10 s, the ICHD characteristic was activated, enabling internal buffering of the gyroscope data (while real-time streaming continues) for subsequent processing. Once a full buffer of data is acquired (1024 data points), the ICHD algorithm starts processing and, in the current implementation (which lacks an RTOS for concurrent code execution), we pause the real-time streaming. The current consumption in this case is about 3.9 mA, at the nominal supply voltage of 3.7 V. Processing time is 1.5 s, which compared to the $1024/F_S$=9.85 s window corresponds to a CPU utilization of 15.4 %, proving that the microcontroller can easily cope with the computational requirements of ICHD.

The average current consumption of the whole device in these conditions (including streaming of raw data) is just 1.76 mA, which means that the 250 mAh battery can power the system for nearly 6 days of continuous monitoring. By disabling the raw data streaming further savings can be achieved, with the average current reaching 1.46 mA and the expected runtime thus exceeding 1 week.

A second test was performed to verify the accuracy of the implemented algorithm, since it only uses 32 bits floating point numbers instead of the double precision, 64-bits arithmetic of the MATLAB reference implementation. To this end, a simple movement consisting of repeated arm swings was performed with the sensor attached to the hand, so as to achieve a nearly periodic, and thus simply to understand by visual inspection, signal. This is indeed shown in Figure 4 (green trace).

For reference, the same figure also shows the instantaneous amplitudes and frequencies as extracted by MATLAB from the streamed raw data. It is apparent that A_1 tracks the amplitude of the signal, while the second component (likely a high-frequency vibration due to the device attachment) is very low. The red bars denote the median f_1 as reported by the device over the ICHD BLE characteristic, that corresponds to the instantaneous frequency computed by MATLAB. (f_1 was chosen because it allows simple visual inspection of its correctness by counting the number of periods in the 10 s interval.)

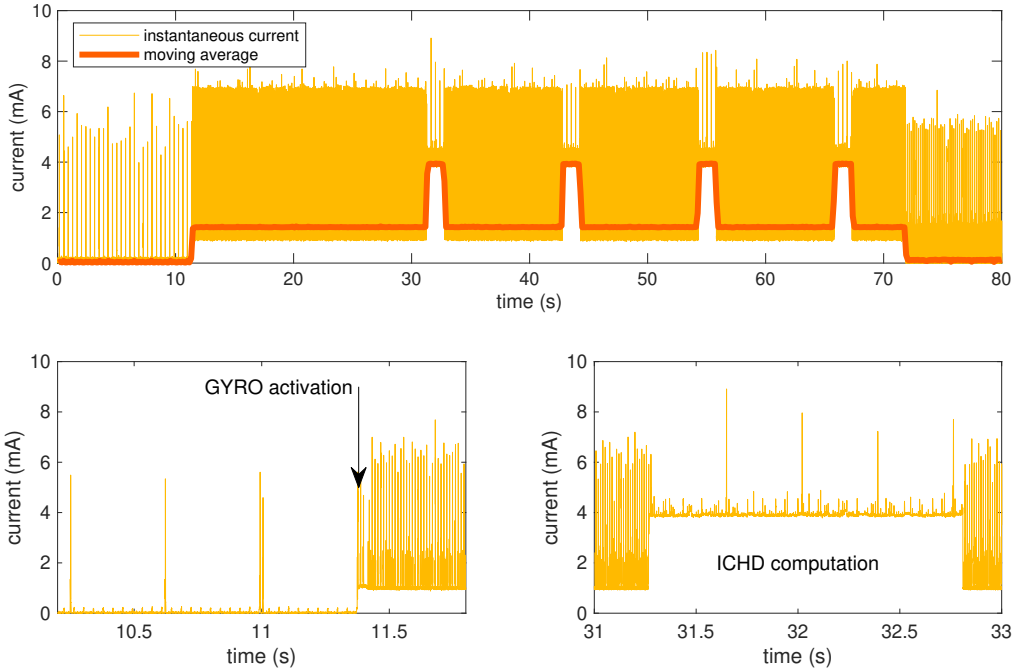

Fig. 3. Measured device current consumption during data acquisition, streaming, and elaboration, at a 3.7 V supply voltage. For the first 10 s or so the device is just connected over BLE to the smartphone, current spikes denote radio activity for keeping the connection alive. When the gyroscope is activated at time 11.4 s the baseline current consumptions grows due to the sensor power draw, and radio communication intervals are shortened to stream the real-time data (lower left insert). A few seconds later the ICHD subsystem is activated, it collects 10 s worth of data and then elaborates it. High CPU activity results in the nearly 4 mA current drawn for the 1.5 s required to perform the computations, as can be seen in the lower right panel. In this implementation, streaming is suspended during elaboration. For reference, a few more computation "events" are shown in the upper panel, and then at around 72 s the connection is closed (the final spike train represents the advertising packets the node sends to re-establish the connection if needed, they stop after a few minutes).

To analytically estimate the actual error, features extracted on-device and by MATLAB are reported and compared side-by-side in Table 1. The error is computed as the absolute value of the difference between the two implementations, divided by the result of the MATLAB one, assumed as a reference. As can be seen, the relative error on the first component is very low, 10^{-5} on the amplitude and 10^{-4} on the frequency, not much more than the resolution limit of 32-bit floats. The error is somewhat larger in the second component, because its amplitude is much smaller than that of the first, and being obtained by subtraction it is bound to have lower accuracy. Nevertheless, it is still negligible for any practical application.

4. Conclusions

In this work we presented an optimized implementation of the ICHD algorithm, suitable to extract compact but significant features from a motion signal. It targets a compact, multi-function and low-power wearable platform we previously developed. Results show that reducing the computation accuracy from 64 bits to 32 bits did not significantly affect accuracy on the major components, and the code can run in real time, using only about one sixth of the available CPU time. The average power consumption, including streaming of the raw data over BLE, is just 6.5 mW, which reduces to 5.4 mW if only the ICHD features are transmitted, enabling a small rechargeable battery to power the device for over a week of operation, making it suitable for continuous monitoring of human activities. The computed data is transmitted over BLE and can thus be collected by any smartphone with a simple app for further relaying or elaboration.

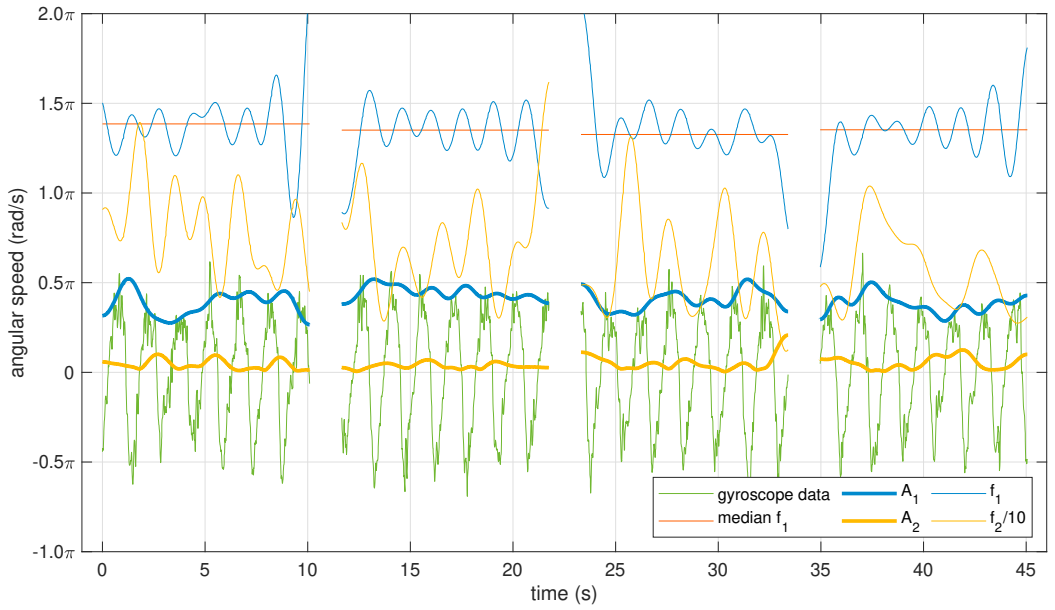

Fig. 4. Example of an acquired signal (gyroscope data) measured with the sensor attached to the hand while swinging the arm, as received from the real-time BLE stream. The "median f_1" line shows the value of the frequency of the first AM-FM component as computed within the device itself and notified over BLE at the end of the elaboration (occurring during the time intervals without data). For reference, the amplitudes A_1, A_2 and frequencies f_1, f_2 of the first two AM-FM components as computed off-line by MATLAB are also shown.

Table 1. Median values of the amplitudes and frequencies of the first two AM-FM components extracted from a sample gyroscope signal (a portion of which is shown in Fig. 4). The acquired signal is an angular velocity so both its amplitude and its (variation) frequency are reported in radians per second. The values computed on-line by the device itself (EMGyro2) and later off-line with MATLAB are reported for comparison.

interval	$\overline{A_1}$ (rad/s)		$\overline{f_1}$ (rad/s)		$\overline{A_2}$ (rad/s)		$\overline{f_2}$ (rad/s)	
	EMGyro2	MATLAB	EMGyro2	MATLAB	EMGyro2	MATLAB	EMGyro2	MATLAB
#1	1.2517	1.2518	4.3519	4.3519	0.1132	0.1132	25.3043	25.3026
#2	1.3716	1.3716	4.2432	4.2426	0.0946	0.0946	21.8441	21.8458
#3	1.2618	1.2618	4.1666	4.1667	0.1313	0.1313	16.3060	16.2873
#4	1.1949	1.1949	4.2490	4.2482	0.1556	0.1557	16.6359	16.6046
#5	1.1417	1.1417	4.2616	4.2616	0.1527	0.1530	18.0039	17.9924
#6	0.8854	0.8854	4.2413	4.2398	0.1056	0.1056	21.0020	21.0021
error	1.03×10^{-5}		4.20×10^{-4}		1.15×10^{-4}		6.37×10^{-4}	

References

[1] M. Alessandrini, G. Biagetti, P. Crippa, L. Falaschetti, C. Turchetti, Recurrent neural network for human activity recognition in embedded systems using PPG and accelerometer data, Electronics 10 (14) (2021).

[2] G. Biagetti, P. Crippa, L. Falaschetti, C. Turchetti, A multi-channel electromyography, electrocardiography and inertial wireless sensor module using Bluetooth low-energy, Electronics 9 (6) (2020).

[3] G. Biagetti, P. Crippa, L. Falaschetti, S. Luzzi, C. Turchetti, Recognition of daily human activities using accelerometer and sEMG signals, in: Intelligent Decision Technologies 2019: Proc. 11th International KES Conference on Intelligent Decision Technologies, Vol. 143, Springer, Singapore, 2019, pp. 37–47.

[4] A. De Vita, G. D. Licciardo, L. D. Benedetto, D. Pau, E. Plebani, A. Bosco, Low-power design of a gravity rotation module for HAR systems based on inertial sensors, in: IEEE 29th Int. Conf. Application-specific Systems, Architectures and Processors, Milan, Italy, 2018, pp. 1–4.

[5] H. Yu, S. Cang, Y. Wang, A review of sensor selection, sensor devices and sensor deployment for wearable sensor-based human activity recog-

nition systems, in: 2016 10th International Conference on Software, Knowledge, Information Management Applications (SKIMA), Chengdu, China, 2016, pp. 250–257.
[6] A. Bacà, G. Biagetti, M. Camilletti, P. Crippa, L. Falaschetti, S. Orcioni, L. Rossini, D. Tonelli, C. Turchetti, CARMA: A robust motion artifact reduction algorithm for heart rate monitoring from PPG signals, in: 23rd European Signal Processing Conference (EUSIPCO), Nice, France, 2015, pp. 2696–2700.
[7] G. Biagetti, P. Crippa, A. Curzi, S. Orcioni, C. Turchetti, Analysis of the EMG signal during cyclic movements using multicomponent AM-FM decomposition, IEEE Journal of Biomedical and Health Informatics 19 (5) (2015) 1672–1681.
[8] J. Tian, P. Zhou, F. Sun, T. Wang, H. Zhang, Wearable IMU-based gym exercise recognition using data fusion methods, in: The Fifth International Conference on Biological Information and Biomedical Engineering, BIBE2021, Association for Computing Machinery, New York, NY, USA, 2021.
[9] D. Naranjo-Hernández, L. M. Roa, J. Reina-Tosina, M. A. Estudillo-Valderrama, SoM: A smart sensor for human activity monitoring and assisted healthy ageing, IEEE Transactions on Biomedical Engineering 59 (11) (2012) 3177–3184.
[10] D. Rodriguez-Martin, A. Samà, C. Perez-Lopez, A. Català, J. Cabestany, A. Rodriguez-Molinero, SVM-based posture identification with a single waist-located triaxial accelerometer, Expert Systems with Applications 40 (18) (2013) 7203–7211.
[11] A. Mannini, S. S. Intille, M. Rosenberger, A. M. Sabatini, W. Haskell, Activity recognition using a single accelerometer placed at the wrist or ankle, Medicine and Science in Sports and Exercise 45 (11) (2013) 2193–2203.
[12] C. Torres-Huitzil, M. Nuno-Maganda, Robust smartphone-based human activity recognition using a tri-axial accelerometer, in: 2015 IEEE 6th Latin American Symposium on Circuits Systems, Montevideo, Uruguay, 2015, pp. 1–4.
[13] F. Miao, Y. He, J. Liu, Y. Li, I. Ayoola, Identifying typical physical activity on smartphone with varying positions and orientations, BioMedical Engineering Online 14 (1) (2015).
[14] D. Anguita, A. Ghio, L. Oneto, X. Parra, J. L. Reyes-Ortiz, Energy efficient smartphone-based activity recognition using fixed-point arithmetic, Journal of Universal Computer Science 19 (9) (2013) 1295–1314.
[15] S. Dernbach, B. Das, N. C. Krishnan, B. L. Thomas, D. J. Cook, Simple and complex activity recognition through smart phones, in: 8th International Conference on Intelligent Environments, Guanajuato, Mexico, 2012, pp. 214–221.
[16] A. M. Khan, Y.-K. Lee, S. Y. Lee, T.-S. Kim, Human activity recognition via an accelerometer-enabled-smartphone using kernel discriminant analysis, in: 2010 5th International Conference on Future Information Technology, Busan, Korea (South), 2010, pp. 1–6.
[17] G. Biagetti, P. Crippa, L. Falaschetti, S. Orcioni, C. Turchetti, An efficient technique for real-time human activity classification using accelerometer data, in: Intelligent Decision Technologies 2016: Proceedings of the 8th KES International Conference on Intelligent Decision Technologies – Part I, Springer International Publishing, Cham, Switzerland, 2016, pp. 425–434.
[18] X. Hu, S.-L. L. Peng, W.-L. L. Hwang, EMD revisited: A new understanding of the envelope and resolving the mode-mixing problem in AM-FM signals, IEEE Transactions on Signal Processing 60 (3) (2012) 1075–1086.
[19] R. Mabrouki, B. Khaddoumi, M. Sayadi, R peak detection in electrocardiogram signal based on a combination between empirical mode decomposition and Hilbert transform, in: 2014 1st International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), 2014, pp. 183–187.
[20] H. Xie, Z. Wang, Mean frequency derived via Hilbert-Huang transform with application to fatigue EMG signal analysis, Computer Methods and Programs in Biomedicine 82 (2) (2006) 114–120. doi:http://dx.doi.org/10.1016/j.cmpb.2006.02.009.
[21] P. Maragos, J. Kaiser, T. Quatieri, On amplitude and frequency demodulation using energy operators, IEEE Transactions on Signal Processing 41 (4) (1993) 1532–1550.
[22] A. Bovik, P. Maragos, T. Quatieri, AM-FM energy detection and separation in noise using multiband energy operators, IEEE Transactions on Signal Processing 41 (12) (1993) 3245–3265.
[23] S. L. Hahn, Hilbert Transforms in Signal Processing, Artech House, Boston, MA, 1996.
[24] Y. Bar-Ness, F. Cassara, H. Schachter, R. DiFazio, Cross-coupled phase-locked loop with closed loop amplitude control, IEEE Transactions on Communications 32 (2) (1984) 195–199.
[25] B. Santhanam, P. Maragos, Multicomponent AM-FM demodulation via periodicity-based algebraic separation and energy-based demodulation, IEEE Transactions on Communications 48 (3) (2000) 473–490.
[26] F. Gianfelici, G. Biagetti, P. Crippa, C. Turchetti, Asymptotically exact AM-FM decomposition based on iterated Hilbert transform, in: Proc. Interspeech'2005 - Eurospeech - 9th European Conference on Speech Communication and Technology, Lisbon, Portugal, 2005, pp. 1121–1124.
[27] F. Gianfelici, G. Biagetti, P. Crippa, C. Turchetti, AM-FM decomposition of speech signals: An asymptotically exact approach based on the iterated Hilbert transform, in: Proc. 2005 IEEE/SP 13th Workshop on Statistical Signal Processing (SSP '05), Bordeaux, France, 2005, pp. 333–337.
[28] S. Gazor, R. R. Far, Adaptive maximum windowed likelihood multicomponent AM-FM signal decomposition, IEEE Transactions on Audio, Speech, and Language Processing 14 (2) (2006) 479–491.
[29] M. Feldman, Time-varying vibration decomposition and analysis based on the Hilbert transform, Journal of Sound and Vibration 295 (3-5) (2006) 518–530. doi:http://dx.doi.org/10.1016/j.jsv.2005.12.058.
[30] F. Gianfelici, G. Biagetti, P. Crippa, C. Turchetti, Multicomponent AM-FM representations: An asymptotically exact approach, IEEE Transactions on Audio, Speech and Language Processing 15 (3) (2007) 823–837.
[31] F. Gianfelici, C. Turchetti, P. Crippa, Multicomponent AM-FM demodulation: The state of the art after the development of the iterated Hilbert transform, in: Proc. 2007 IEEE International Conference on Signal Processing and Communications (ICSPC 2007), IEEE, Dubai, UAE, 2007, pp. 1471–1474.
[32] G. Biagetti, P. Crippa, L. Falaschetti, S. Orcioni, C. Turchetti, Human activity recognition using accelerometer and photoplethysmographic signals, Smart Innovation, Systems and Technologies 73 (2018) 53–62.
[33] X. Hu, S. Peng, B. Guo, P. Xu, Accurate AM-FM signal demodulation and separation using nonparametric regularization method, Signal

Processing 186 (2021).

[34] C. Carranza, V. Murray, M. Pattichis, E. S. Barriga, Multiscale AM-FM decompositions with GPU acceleration for diabetic retinopathy screening, in: 2012 IEEE Southwest Symposium on Image Analysis and Interpretation, 2012, pp. 121–124. doi:10.1109/SSIAI.2012.6202468.

[35] Y.-Y. Hong, Y.-Q. Bao, Fpga implementation for real-time empirical mode decomposition, IEEE Transactions on Instrumentation and Measurement 61 (12) (2012) 3175–3184. doi:10.1109/TIM.2012.2211460.

[36] G. Biagetti, P. Crippa, L. Falaschetti, M. Alessandrini, C. Turchetti, Wearable acceleration-based human activity recognition using AM-FM signal decomposition, Intelligent Decision Technologies 2022 - Proceedings of the 14th KES International Conference on Intelligent Decision Technologies (KES-IDT-22), (in press) (June 2022).

[37] G. Biagetti, P. Crippa, L. Falaschetti, A. Mansour, C. Turchetti, Energy and performance analysis of lossless compression algorithms for wireless EMG sensors, Sensors 21 (15) (2021).