A machine learning approach for automated wide-range frequency tagging analysis in embedded neuromonitoring systems

(Article begins on next page)

03 May 2024

A Machine Learning Approach for Automated Wide-Range Frequency Tagging Analysis in Embedded Neuromonitoring Systems

Fabio Montagna[a,*], Marco Buiatti[b], Simone Benatti[a,*], Davide Rossi[a,*], Elisabetta Farella[c], Luca Benini[a,d]

[a]*Energy Efficient Embedded Systems (EEES) Lab - DEI, University of Bologna*
[b]*Center for Mind/Brain Sciences, University of Trento*
[c]*Energy Efficient Embedded Digital Architectures (E3DA) Unit - ICT Center, Fondazione Bruno Kessler*
[d]*Integrated System Laboratory ETH, Zurich*

Abstract

EEG is a standard non-invasive technique used in neural disease diagnostics and neurosciences. Frequency-tagging is an increasingly popular experimental paradigm that efficiently tests brain function by measuring EEG responses to periodic stimulation. Recently, frequency-tagging paradigms have proven successful with low stimulation frequencies (0.5 - 6 Hz), but the EEG signal is intrinsically noisy in this frequency range, requiring heavy signal processing and significant human intervention for response estimation. This limits the possibility to process the EEG on resource-constrained systems and to design smart EEG based devices for automated diagnostic. We propose an algorithm for artifact removal and automated detection of frequency tagging responses in a wide range of stimulation frequencies, which we test on a visual stimulation protocol. The algorithm is rooted on machine learning based pattern recognition techniques and it is tailored for a new generation parallel ultra low power processing platform (PULP), reaching performance of more that 90% accuracy in the frequency detection even for very low stimulation frequencies (< 1 Hz) with a power budget of 56 mW.

Keywords: EEG; BCI; Machine learning; Frequency-tagging; SVM; Embedded systems

1. Introduction

Neurological disorders affect nearly one billion people, a considerable percentage of the world population [1]. The estimated economic costs were more than 2 trillions of USD in 2010 [2], with a high social impact. Despite the fact that effective treatments are available, a significant part of the population is untreated, because of inadequate healthcare infrastructure, lack of trained staff, effective diagnostics and screening tools. Among diagnostic and screening techniques, Electroencephalography (EEG) analysis and instrumentation is an established standard, since it directly records the electrical field generated by neural activity with a set of electrodes distributed on the head surface (scalp) [3]. Thanks to its effectiveness, non-invasiveness, low cost and portability, EEG is one of the most used techniques for investigating brain function and pathology, both in clinical settings and scientific research [4] [5] [6].

One effective use of the EEG signals is to analyze brain responses to specific stimuli. The most popular method to measure the EEG response to a stimulus is to average the EEG signal across several stimulus presentations (Event-Related Potential, ERP [7]). Since the neural activity unrelated to the stimulus typically fluctuates within the same time scales of the stimulus related activity, a high number of stimulus presentations is needed to average it out and extract the stimulus-related response.

An alternative technique, frequency-tagging (FT), has been developed to reliably measure stimulus-related EEG responses in a much shorter time. This technique exploits the property of the brain activity to respond to a visual or auditory stimulus presented periodically at a specific (i.e. "tag") temporal frequency by resonating at the same frequency during the stimulation period [8, 9].

---

*Corresponding Author

**Fabio Montagna, Marco Buiatti and Simone Benatti contibuted equally to this work.

Email addresses: `fabio.montagna@unibo.it` (Fabio Montagna), `marco.buiatti@unitn.it` (Marco Buiatti), `simone.benatti@unibo.it` (Simone Benatti), `davide.rossi@unibo.it` (Davide Rossi), `efarella@fbk.eu` (Elisabetta Farella), `luca.benini@unibo.it`, `lbenini@iis.ee.ethz.ch` (Luca Benini)

This effect is manifested in the EEG recordings by a sharp peak in the power spectrum of the signal at that specific tag frequency. Since the EEG ongoing activity is broad-band in frequency, the stimulus-related response in the frequency domain is very easily discriminated from the stimulus-unrelated activity, yielding a much higher SNR than the one obtained with ERPs. Moreover, since most EEG artifacts (eye movements, blinks) are also broad-band in frequency, FT is more robust than ERP to artifacts and requires a lighter artifact rejection procedure.

Thanks to the short time needed to have a reliable response, FT has always been used in clinical settings to test the integrity of sensory areas [3] in the visual domain (classically defined Steady-State Visually Evoked Potentials - SSVEP [9]) and in the auditory domain (classically defined auditory steady-state responses - ASSR [8]), by presenting simple visual and auditory stimuli at relatively high frequencies (10 to 40 Hz), since sensory systems are most responsive at those frequencies. Since the amplitude of both ongoing EEG activity and eye-movement-related artifacts are relatively low in this frequency range, fast, automatic techniques have been developed to rapidly extract the frequency responses, in particular in the field of SSVEP-based BCI systems [10].

Most recently, the use of FT has been extended to investigate the neural responses related to higher-level perceptual or cognitive functions, such as attention [11], speech [12] or face [13] recognition. Since such functions require longer neural processing, stimulation frequencies in a low-frequency range (0.5-6Hz, hereafter referred to as "low-frequency") have to be used. It has been shown that FT is still effective at those frequencies [12]. Because of their performance in obtaining a stimulus-specific neural response in a short time, very recent FT designs based on stimulations in the low-frequency range have been successfully used as a tool for investigating the neural basis of cognitive development in very young children [14], [15]. Given these positive results, FT is a promising tool for testing brain function in clinical settings and/or with vulnerable populations as newborns or aged people.

However, for FT designs based on the low-frequency range, EEG ongoing fluctuations and artifacts are much more relevant than in the typical frequency range of SSVEP (>6Hz). Therefore a significant human intervention is needed to clean the data from artifacts and extract the response. Typically, the EEG traces are acquired and processed off-line on benchtop platforms, since most of the techniques to analyze brainwaves requires heavy computational processing and visual in-

spection from technicians or medical staff. This procedure implies significant cost and time for an accurate analysis, due to the need for data transfer, off-line visualization, manual inspection and tagging. For instance, artifacts identification and removal requires a combination of the visual evaluation of the EEG trace and of algorithmic techniques such as Independent Component Analysis (ICA), digital filtering, interpolations and averaging. Automating these analyses for FT stimulations in the low-frequency range would dramatically improve research and diagnosis, enabling the design of extensive screening systems for many neural disorders.

Luckily, we are witnessing the massive technological trend of embedded wearable applications, that are quickly becoming pervasive [16], led by the constant growth of the healthcare market and by the boost of digital technologies integration [17]. This trend paves the way for designing embedded, energy-efficient systems for biosignal processing based on advanced algorithmic techniques [18, 19, 20].

This work presents a framework for automated analysis of FT responses in EEG traces from FT stimulation protocols, with the specific purpose of application both in the standard frequency range (>6Hz) and in the low-frequency range (0.5-6Hz) of FT stimulation. The algorithm, using machine learning techniques, automatically removes the EEG segments affected by artifacts of various origin and detects the presence of the resonant frequency even at very low (< 1Hz) stimulation frequencies, for which detection is typically very challenging. We tested the algorithm on 4 subjects undergoing a visual stimulation with an on-off checkerboard pattern at 0.667Hz, 0.8Hz, 4Hz and 12.5Hz, showing that automated processing is able to detect the presence of resonance at all the target frequencies. Results are compared with the manual processing performed by an expert human. Furthermore, we show the implementation and the profiling of the algorithm on a programmable ultra low power multicore platform (PULP), demonstrating that it is possible to design a fully wearable system for autonomous on line detection of resonant frequencies during *in vivo* tests featuring a power consumption of 56mW leading to a battery-life of almost 24h.

2. Related Work

Processing of EEG data is a complex and multiple steps process involving filtering, feature extraction, machine learning and classification techniques [31]. A general structure of BCI system is presented in [32], showing a series of functional processing blocks among which signal pre-processing, artifact removal, feature

Table 1: Comparison with state-of-the-art of Artifact Removal.

	Barbati [21]	Delorme [22]	Mantini [23]	Li [24]	Viola [25]	Joyce [26]	Okada [27]	Mognon [28]	**This Work**
STIMULI	-	visual	auditory,visual	visual	auditory,visual	-	visual	auditory,visual	visual
HUMAN INTERVENTION	✓	✓	✓	✓	✓	✓	✓	x	x
ICA	✓	✓	✓	✓	✓	✓	✓	✓	x
DOMAIN	time	time	time	space	space	both	both	both	both
EMBEDDED	x	x	x	x	x	x	x	x	✓
REAL-TIME	x	x	x	x	x	x	x	x	✓
NUMBER OF ELECTRODES	28	32	153	32	30-128	6	306	64	64

Table 2: Comparison with state-of-the-art of Frequency Tagging.

	Buiatti [12]	Ding [29]	Baldauf [30]	Kim [11]	Rossion [13]	Karbdebon [14]	de Heering [15]	**This Work**
STIMULI	auditory	auditory	visual	visual	visual	auditory	visual	visual
METHOD	FT	FT	FT	FT	FT	FT	FT	FT
FREQUENCY RANGE [Hz]	1.4-4.2	1-4	1-2	12.5-16.67	1.18-16.37	1.39-4.17	1.2-6	0.667-12.5
EMBEDDED	x	x	x	x	x	x	x	✓
NUMBER OF ELECTRODES	129	157	-	59	128	64	32	64

extraction that can be preceded by signal enhancement and accompanied by dimensionality reduction, feature selection, classification and post-processing. In case a direct real-time feedback is required, further processing steps are considered such as algorithms for actuator control, tuning for adaptive and smart feedback, etc. The complexity, the non-automatization and the number of processing steps are some of the barriers to the design and use of wearable EEG solutions in daily life [33].

It is convenient to tailor the choices of the algorithm for artifact removal, as well as those for the following steps, such as feature extraction and classification, to the target application. As described in the Introduction, here we focus on experimental protocols based on FT designs, where the response to be detected is determined by the stimulation frequency. In this case, the influence of artifacts on the response depends on the stimulation frequency. Since most artifacts (blinks, eye movements, heart beats) are characterized by a broad frequency range mainly on frequencies lower than approximately 6Hz, the response to visual or auditory periodic stimulation at frequencies higher than 6 Hz (the standard frequency range of SSVEP) is easily detectable with very light artifact correction, and efficient techniques for on-line processing of SSVEP-based BCI already exist (e.g. [10]). Conversely, for stimulation frequencies in the low-frequency range (0.5-6Hz), accurate artifact correction/removal is fundamental for the correct identification of the frequency-tagged response, although artifact rejection thresholds are generally 2-3 times higher than for ERP designs [12]. Despite the

recent rising of studies using FT designs in the low-frequency range [12] [29] [30], artifact correction/removal employed in these studies is usually the same used in ERP designs with lighter thresholds, and no tool specific for low-frequency FT design exists in the literature.

Several solutions have been proposed to perform automatic EEG artifact removal. For example, ICA, that we already mentioned as one of the most efficient methods for artifact identification, has been subject to several attempts of semi-automatization in time [21] [22] [23] and space domain [24] [25] or both domains [26] [27]. An interesting fully automatic approach is presented by Mognon et al. [28], based on the automatic adjustments of the algorithm's parameters to the data used for the spatial and temporal feature extraction for IC classification. However, this method still requires to process data in two domains, time and space, and fuse results, and is uniquely based on the computationally intensive ICA decomposition. It therefore presents challenges for a resource constrained device and it can only be performed off-line. Table 1 shows a summary of comparison between the state of the art of artifact removal solutions. Table 2 present the state of the art in FT solutions. The comparison covers the most relevant characteristics mentioned in the references.

Concerning the hardware requirements, most of the previously mentioned algorithms are meant to run on desktop machines or servers, eventually equipped with real-time monitoring tools which synchronize through a USB cable with the helmet for EEG tracks visual-

ization, while data analysis is often performed off-line using EEG feature extraction and interpretation toolboxes such as EEGLAB [34] or OpenViBE [35]. While most traditional EEG monitoring systems used today in hospitals are stationary, wired, and cumbersome systems, several clinical applications can benefit from intelligent wearable, wireless, convenient, and comfortable solutions that provide high signal quality. Transferring the technology used in hospitals to users homes would allow a large scale deployment of such clinical applications, significantly improving the therapies and knowledge by increasing the amount of data that can be collected and analyzed by the experts. For this reason, some portable EEG acquisition and monitoring systems have been recently developed, mainly meant for raw data transmission through wireless communication stacks [36]. Notable examples of recent wireless EEG monitoring systems are offered by Quasar [37], IMEC [38], Emotiv [39], NeuroSky [40]. However, these systems are only meant for consumer applications featuring 2 to 8 electrodes; the only one featuring a reasonable number of electrodes suitable for medical applications is g.tech [41].

In these systems tiny micro-controllers (MCUs) such as PIC-32 [42] or Cypress CY8C38 [36] are only used for transmission setup and control purposes. Therefore, due to the limited computational capability of the processors, these devices are only capable of transmitting raw EEG data, leading to a huge amount of wireless bandwidth. While Bluetooth (BT) is widely used for data transfer, the sheer amount of raw data transmitted during an EEG recording makes this protocol quite power hungry, especially for systems with a large number of electrodes (e.g., 64). Alternative more energy efficient protocols can be considered, such as Bluetooth Low Energy (BLE), even if at the price of lower throughputs and higher latency, therefore not always adequate for physiological data streaming [43] [44]. However, the use of a powerful and energy efficient processor into the system enables processing of data before transmission and consequent bandwidth reduction. In fact, only relevant content is sent with lower throughput, allowing the use of low-power protocols.

The lesson learned from the analysis of the literature is that the design of an energy efficient embedded system for real time EEG processing requires a multilevel design. Combining the algorithm and technology approaches, we designed an algorithm tailored for parallel, ultra-low power processing platforms [45][46]. We exploit low-voltage operation and parallel computing to provide more than one order of magnitude of better performance and energy efficiency with respect to tra-
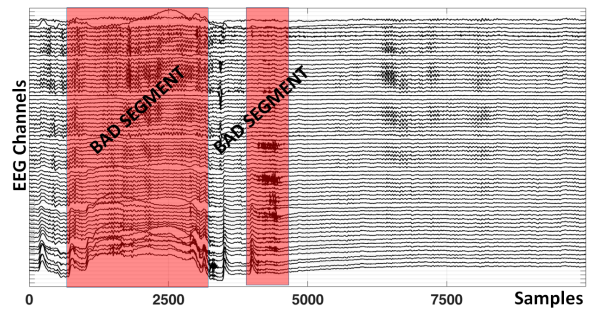

Figure 1: EEG traces with bad segments

ditional MCUs. By exploiting close-to-sensor energy efficient data processing, it is possible to significantly reduce wireless transmission, designing more compact EEG monitoring systems and eliminating the need of an external host (e.g. a PC or a notebook), which can be replaced with a portable device such as a smartphone or a tablet with data visualization or device diagnostics.

3. Materials and Methods

3.1. Stimuli and EEG Acquisition Setup

Four right-handed (Edinburgh Inventory) native Italian speakers participated in the experiment (2 females; 21-29 years, mean age 23.3 years). All participants had normal or corrected-to-normal visual acuity, and reported no history of neurological or psychiatric disorders. All participants provided informed written consent to take part in the experiment, which was approved by the Ethical Committee of the University of Trento (Italy).

Stimuli consisted of a black and white 10x10 square checkerboard subtending approximately 15° by 15° of visual angle, on a uniform grey background, presented at a distance of 80 cm from the subjects eyes. Stimuli were presented with a sinusoidal on-off 100% contrast temporal modulation (black/white squares starting gray as the background, turning black/white at half cycle and ending gray at the end of the cycle) at 4 frequency rates (0.667 Hz, 0.8 Hz, 4 Hz, 12.5 Hz) in blocks of 28 cycles (42s), 32 cycles (40s), 160 cycles (40s), 360 cycles (28.8s), respectively, using the Psychtoolbox 3.0.12 for Windows in Matlab 8.0 (MathWorks Inc.). A sinusoidal contrast modulation was used because it generates fewer harmonics (i.e., responses at exact multiple of the stimulation frequency, reflecting the non linearity of the brain response [9, 47]) and because, since the on-off dynamics is smooth, it is a more pleasant and less fatiguing visual stimulation than a squarewave stimulation mode
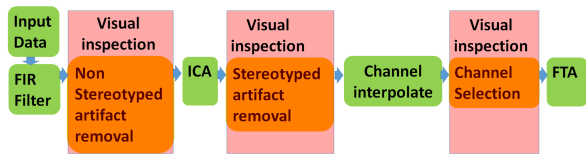
4

Figure 2: Block diagram of the standard analysis steps

for the subjects. Subjects were asked to fixate at the center of a grey diagonal cross overlapped to the checkerboard. Each subject was presented with two series of blocks (presentation rates were randomized within each series).

The experiment was performed in an electrically shielded and sound-attenuated cabin. EEG was recorded with a BrainAmp amplifier (Brain Products, Munich) using 64 Ag/AgCl sintered ring electrodes mounted in an elastic cap (Easycap, Munich) and placed equidistantly according to the 10/20 system [48], with a vertex reference (Cz) and ground electrode in AFz. Electrode impedances were kept below 15kOhm. Data were sampled at 500Hz and analog filtered between 0.016 and 250Hz during recording.

3.2. Standard analysis

3.2.1. Pre-processing and artifact removal

In Fig. 1 a typical EEG trace is shown. Segments affected by artifacts are highlighted under red transparent boxes. Fig. 2 shows the block diagram of the standard analysis, based on visual inspection of the whole EEG trace. Continuous raw data were imported in the EEGLAB software [34] and band-pass filtered between 0.1 and 40Hz with the default EEGLAB filter (a Hamming windowed sinc FIR filter) to remove DC and high-frequency noise. Data were segmented in windows corresponding to the stimulation blocks. Each segment was visually inspected and portions containing non-stereotyped paroxysmal artifacts were discarded. Bad channels containing jumps larger than 200μV were discarded (no more than one per subject). To identify and remove stereotypical artifacts, the default EEGLAB ICA decomposition was computed on the concatenation of all segments. Blinks, eye-movements and other topographically localized artifacts were discarded by removing the corresponding independent components identified by ADJUST, an algorithm for automatic detection of artifacted ICA components [28]. Muscle artifacts were discarded by removing related ICA components identified by visual inspection of their topography and spectro-temporal profile. EEG signals in bad channels were interpolated with the EEG signals from

neighbouring channels (standard spherical interpolation method in EEGLAB), and the resulting clean data were re-referenced to average reference. In summary, this standard procedure requires expert off-line intervention for the identification of bad data segments and artifacted ICA components, and computationally heavy ICA decomposition.

3.2.2. Frequency-Tagging Analysis (FTA)

To obtain a high frequency resolution with one bin centered on the stimulation frequency, for each tag frequency, EEG data from each channel were segmented into epochs of 18s (9000 time points, frequency resolution=0.0556Hz), 20s (10000 time points, frequency resolution=0.05Hz), 10s (5000 time points, frequency resolution=0.1Hz) and 10s (5000 time points, frequency resolution=0.1Hz) for the stimulation frequencies 0.667Hz, 0.8Hz, 4Hz, 12.5Hz, respectively, overlapping for half of their length. For each electrode, the Fourier transform $F_m(f)$ of each epoch was calculated using a Fast Fourier Transform (FFT) algorithm (MATLAB, Natick, MA) on m samples. The power spectrum was calculated from these Fourier coefficients as the average over epochs of the single-epoch power spectrum:

$$P_m(f) = F_m(f)F_m(f) \tag{1}$$

Normalized power (NP) at the tagged frequencies (0.667Hz, 0.8Hz, 4Hz, 12.5Hz, respectively) was calculated as the ratio between the power spectrum at the tagged frequency and the average power spectrum at 2 neighboring frequency bins.

3.3. Automated Analysis

This sub-section describes the proposed machine-learning approach for automated analysis of EEG traces. The algorithm detects frequency peaks correlated to given visual stimula. The detection is composed of 2 parts described in the following: *Artifact Removal* and *Frequency Detection*.

3.3.1. Artifact Removal

In the *artifact removal* section a supervised classification algorithm detects and removes signal segments affected by artifacts that can degrade the analysis. A diagram of the algorithm is shown in Fig. 3. First, the signal is preprocessed using a bandpass FIR filter from 0.1 to 40Hz to remove higher frequencies. After filtering, the data is windowed and signal features are extracted calculating the DWT (Discrete Wavelet Transform) to provide information on the frequency content of the signal in the time domain. Detail Coefficients of the DWT
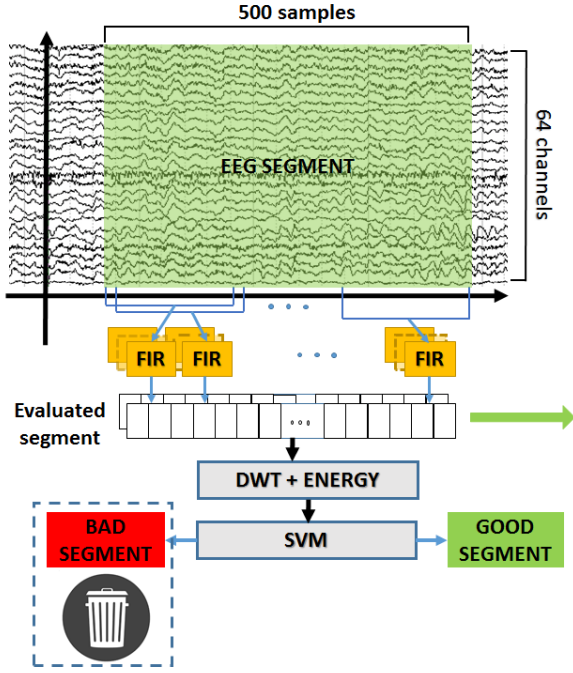
Figure 3: Block diagram of the automated artifact removal

Figure 4: ROC curve for SVM performance evaluation.

are used to extract the energy of the signal. Energy values are calculated at the 4 level of decomposition of the DWT according to (2) resulting in a vector of $1x256$ elements for each level.

$$\mathbf{E}_{D_{(n)}} = \sum_{i=0} \left| (D_{(n)}[i]) \right|^2 \qquad (2)$$

The energy coefficients are the input for a Support Vector Machine (SVM) classifier [49], widely used in biosignal embedded application by virtue of its theoretical robustness and implementation efficiency [50]. The classifier detects if the acquired segments are affected by movement or muscular artifacts and discards them. For the training phase 30% of samples belonging to the starting dataset and the related labels are used. Data is scrambled, downsampled and scaled to improve accuracy during prediction. The remaining samples of the dataset are tested during the classification phase. The SVM has 256 input features, thus we compared linear against Gaussian kernels to find the best tradeoff between accuracy and computational cost. In this application, the accuracy of the two kernels is similar but the linear kernel needs a smaller computational effort, hence we selected it for our embedded implementation [51]. The tuning of the parameters of the classifier is performed calculating the ROC curve [52] obtained
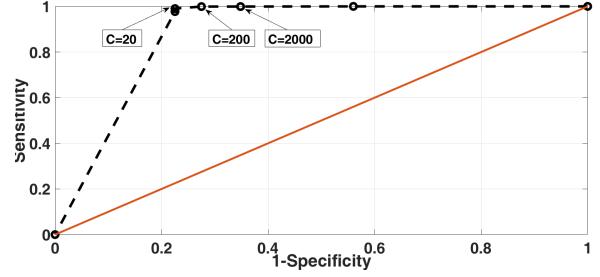
varying the parameter C of the classifier. For small values of C the separation hyperplane presents a smaller-margin while a larger-margin is obtained increasing C values. In Fig. 4 we present the ROC curve with a classification accuracy ranges between 85% to 96% and the best value shown is $C = 20$.

3.3.2. Frequency Tagging Analysis (FTA)

A diagram of the proposed algorithm is shown in Fig. 5. As in the standard analysis, data derived from the Artifact Removal stage are segmented into epochs of 18s (9000 time points, frequency resolution = 0.0556 Hz), 20 s (10000 time points, frequency resolution = 0.05 Hz), 10 s (5000 time points, frequency resolution = 0.1 Hz) and 10 s (5000 time points, frequency resolution = 0.1 Hz) for the stimulation frequencies 0.667 Hz, 0.8 Hz, 4 Hz, 12.5 Hz, respectively, overlapping for half of their length.

The signals in which we are interested are generated by the visual cortex, located in the occipital region of the head. Hence, to extract the frequency information among the sensors placed in this region we apply the Principal Component Analysis (PCA), an orthogonal transformation that converts possibly correlated data distributed on p sensors into a set of linearly uncorrelated components ($l < p$). This algorithm is widely used in neural processing [53], to extract maximum variance components from a dataset or for dimensionality reduction [54] converting an input matrix into a new coordinates system through a linear transformation.

In this application, we extract the maximum variance of the data from the 8 occipital EEG channels of the 10/20 system [48] maintaining 3 components with more than the 90% (to ensure this condition to be always satisfied, the number of PCs is fixed to 3) of the variance of the original input data.

This step of the processing chain has the double advantage to extract the frequency information among several sensors and to reduce the memory requirements
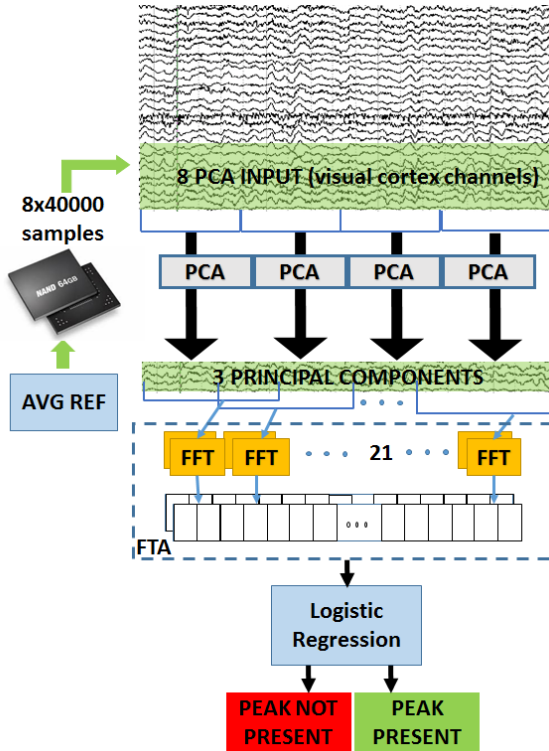
6

Figure 5: Block diagram of the automated frequency detection

Machine learning provides robust solutions such as logistic regression [55]; this algorithm is used to detect the presence of a frequency peak.

The features used in this algorithm are the amplitude of the frequency peak, the normalized ratio between neighbours coefficients (i.e. the coefficients adjacent to the target frequency) and the ratio between the peak and the sum of the left and right neighbours, to evaluate if the peak is localized and narrowband. Formulas of the 3 features are shown below, in (3)

$$
\begin{aligned}
x_0 &= p_n \\
x_1 &= \frac{p_n}{|p_{n-1} - p_{n+1}| + \varepsilon_d} \\
x_2 &= \frac{p_{n-1} + p_{n+1}}{p_n}
\end{aligned}
\tag{3}
$$

where p_n is the target bin, p_{n-1} and p_{n+1} are the previous and the next bin respectively and ε_d is a given constant to prevent division by zero. Even in this case the classifier is trained off-line, once, using a training dataset. The weight parameters θ are obtained by application of the gradient descent technique, an iterative method to minimize convex functions. In this application, the peak recognition based on the LR reaches 92% accuracy. This accuracy values are aligned with SoA of machine learning applications for biomedical purposes [56, 57, 58].

3.4. PULP platform

With the aim of implementing a complex processing chain that implies a considerable computational effort at energy levels compatible with a wearable setup, PULP is chosen as reference platform. PULP[1] is an ultra-low-power parallel computing platform targeting the performance and power requirements of several emerging near-sensor processing applications, such as low-power image, video and audio analytics, and processing of biometrical signals. By exploiting near-threshold operation, advanced low-power FD-SOI technology and an architecture carefully tuned for low-power operation, PULP can deliver several hundreds of MOPS within a power envelope of up to few tens of mW [46], satisfying the requirements of these applications both in terms of performance and power. Fig. 6(b) shows the die micrograph of the third embodiment of the PULP platform that was used for characterization of the performance

of the computational framework, making it suitable for embedded implementation. The FT analysis is applied to the three components extracted by the PCA. This is a powerful method that allows to maximize variance in the first Principal Components through a linear transformation. Performing FTA on each channel shows that the target frequency is not always present or clear in all channels of the visual cortex and in this case they usually show small magnitude peaks. This means that clinicians should manually inspect the traces and select the best channels, i.e. the closest to the EEG response, to obtain a clear peak that allows to make an accurate diagnosis. PCA shows that the peak at the target frequency is always present and visible in the first PC and the magnitude is higher (up to 10 times in the 0.667Hz) compared to the peak obtained considering only data acquired from one channel. In this way, the computational effort dramatically decreases because FTA is performed only on three components rather than on the whole channels of the occipital region.

Since signals and FFT coefficients are affected by high variability caused by physiological or setup-dependent reasons, a simple threshold detection does not guarantee adequate robustness for this application.
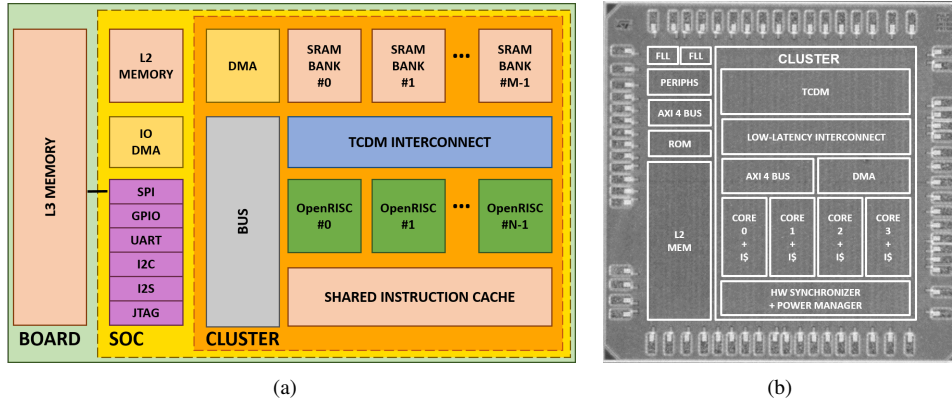
Figure 6: A general view of PULP Architecture (a) and the layout of the PULPv3 chip used for performance and power characterization (b).

and power models, while the system on chip architecture used in this work is reported in Fig. 6(a) and described below.

PULP is a cluster with a parametric number of processors (2-16). The processors used in the cluster are based on an optimized implementation of the open-source OpenRISC Instruction Set Architecture (ISA), and they feature micro-architectural optimizations and instruction extensions targeting energy efficient digital signal processing. The ISA extensions include zero-overhead hardware loops, load and store operations with automatic pointers increment and floating-point units [59], necessary to deal with applications requiring high precision and high dynamic range such those in the field of EEG data processing.

The L1 data memory is composed of 64kB of multi-banked Tightly Coupled Data Memory (TCDM) acting as software-managed scratchpad memory. The 512kB off-cluster L2 memory can be accessed by a tightly coupled DMA optimized for low power through the 64-bit AXI4 interconnect, which guarantees high L1 to L2 communication bandwidth (i.e. up to 32Gbit/s at 500MHz). The cluster and the rest of the SoC reside in two clock and power domains controlled by Frequency-Locked Loops (FLLs) and external voltage regulators. Hence, voltage and frequency can be scaled according to the performance requirements of the applications.

The SoC features a wide set of peripherals, including I2C, I2S, UART, GPIOs, a JTAG port for debug, a (quad) SPI slave, and a (quad) SPI master, which enables data transfers as fast as 400Mbit/s toward external SRAMs or FLASH memories acting as the third level of memory hierarchy. For this purpose, a non-volatile Cypress CY15B104Q ferroelectric RAM (FRAM) is employed as external L3 memory. It is connected to

the chip through a QSPI and can reach 400Mbit/s as maximum bandwidth, assuming that the QSPI works at a frequency equal to 100MHz. Power consumption of L3 memory goes from 16.2mW in active mode to 0.33mW in sleep mode. An I/O DMA subsystem allows to autonomously copy data between the L2 memory and the external interfaces, even when the cluster is power gated. This feature allows to relieve cores from the frequent control of peripherals necessary in most of commercial micro-controllers, and to implement a double buffering mechanism both between IOs and L2 memory and between L2 memory and TCDM. Therefore, I/O transfers, L2 memory to TCDM transfers, and computation phases can be fully overlapped within the PULP SoC, hiding the latency of transfers from slow peripherals typical of micro-controllers.

The PULP platform relies on OpenMP 3.0 parallel library that operates on top of GCC 4.9 toolchain for programming. The OpenMP implementation is based on a highly optimized bare-metal library [60], which avoids the presence of an operating system that would introduce huge software overheads not suitable for ultra-low-power parallel accelerators. The PULP platform features a set of software tools that include a virtual platform and support for parallel profiling, useful to implement and debug applications that run on the architecture, and estimate their execution time. The toolchain was used in this work to evaluate the parameters of the system, simulating architectural configurations not necessarily implemented on the silicon prototypes, such as the number of processors, and the presence of floating-point units. To estimate the power consumption of the architecture, data were extracted from measurements on the silicon prototypes and adapted to the configurations actually employed in the exploration.
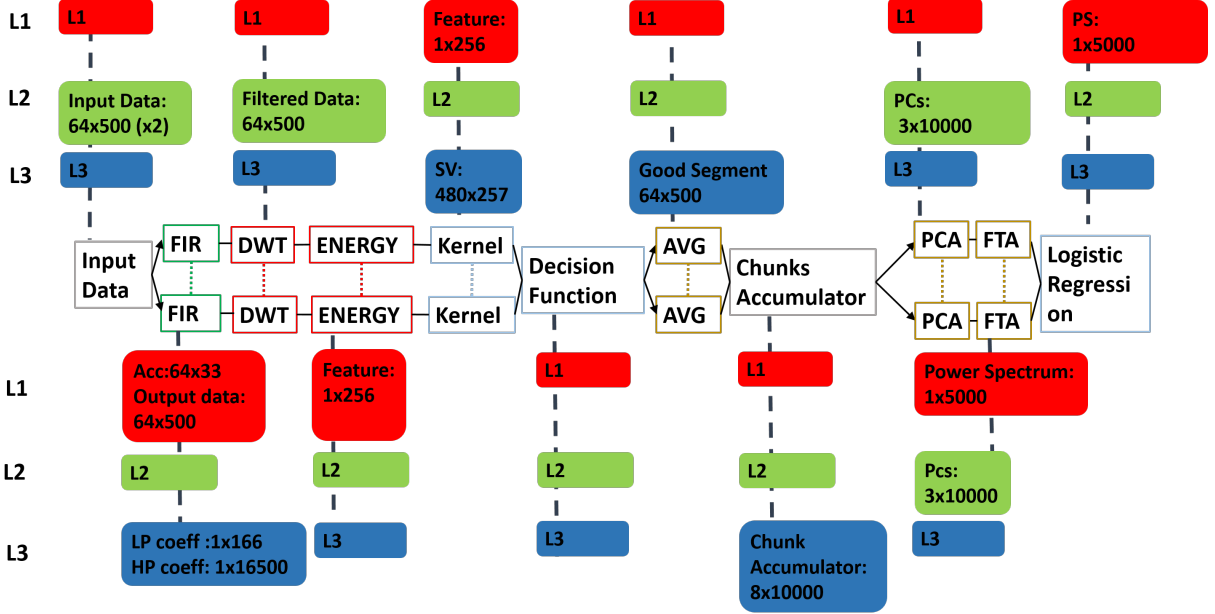
Figure 7: Automated Artifacts Removal and Frequency Detection computational Kernels.

3.5. Implementation on PULP

This section describes the implementation of the processing chain described in Section 3.3 on the PULP architecture. The processing chain was decomposed in 6 sub-kernels, and each kernel was analyzed individually. The level of parallelism that can be achieved depends on the nature of the kernel itself. Fig. 7 shows a block diagram describing the processing chain emphasizing the two key aspects of the proposed implementation on a low-power parallel embedded system: parallelization scheme and memory requirements. In the following, we assume a configuration of the PULP platform with 8 cores and floating point units, unless differently specified.

Sampling data from 64 channels with a sample rate equal to 500Hz produces 64x500 samples per second, which have to be processed in real-time. We have assumed the data to be streamed into the SoC by 8 x 8-channels Texas Instrument ADS1298 ADCs through the SPI master port in 16-bit format, which are then stored in the L2 memory by the IO DMA. As first processing step, a low-pass FIR filter with 166 taps and a high-pass filter with 16500 taps are employed. Hence, before the processing can start, 1s of latency is needed to fill the samples buffer. After the first second of acquisition, a new 64x500 sample window is available every second. Hence, given that the order of the FIR filter is 16500 a delay of 33 seconds is necessary before the first meaningful filtered sample is produced. This leads

an L2 memory requirement of 2 x 125kB to store the current and previous input samples windows, 66kB of L2 memory to store the FIR coefficients (i.e. *High-pass FIR coeff* and *Low-pass FIR coeff*), and 8kB to store the partial accumulations (i.e. *Accumulator*), which can be kept into the L1 memory. Every second, a new set of partial accumulations are computed by the processor, taking as input the current 64x500 input sample window. To overlap the computation and data transfer (L2 to L1) phases, a double-buffering mechanism is employed, with 3 buffers, each one of 16kB (one for the input data, one for the output data, and one for the current data). The FIR filter is parallelized at block level (i.e. every core operates on 8 channels); once a new block of samples is available, threads compute outputs using past values of the filter and the latest samples, and store the accumulated data into the L1 memory buffer. Since the parallel FIR implementation is based on weighted sums without any dependency and the number of channels is a multiple of the number of cores, the workload is always perfectly balanced among cores and there is no need to use synchronization barriers leading to an almost ideal speed-up with respect to the single core execution. A copy of the filtered channels is also stored into the L3 memory, to be eventually re-loaded by the processing chain for frequency detection, if marked as good segment by the artifact removal algorithm.

After filtering the first samples window, DWT and Energy kernels (i.e. feature extraction) can be executed.

Both kernels can be efficiently parallelized on the architecture since each thread can operate independently on a separate channel. Moreover, being the number of channels a multiple of the number of cores, the workload is again perfectly balanced, scales perfectly upon 2, 4 or 8 cores showing nearly ideal speed-ups. The output of the DWT and the Energy kernels are stored in a vector of dimension 1x256, (i.e. *feature*, 1kB) that can be allocated into the L1 memory.

The next processing step is the SVM classification. The main issue with the SVM algorithm is the large amount of memory required to store the support vectors that implement the model (i.e. *SV*). Each SV is composed of 256 values plus a coefficient, thus the total size per support vector is 1kB. The chosen model includes 480 SVs, requiring a storage of 482kB. In the proposed implementation, SVs are permanently stored in the L3 memory. With an SPI, data is transferred from the external L3 to L2 memory and from L2 to L1 with the DMA exploiting a double buffering polity. The parallelization of this kernel is also highly efficient since each thread independently operates on one of the 480 support vectors of the model, and the parallelism is so high (i.e. 480) that workload unbalancing does not impact performance.

Each time a new window of data marked as good segments (i.e. not discarded) is available, the filtered samples are loaded from L3 memory, and the mean value between all the channels is computed and subtracted from all the samples of each channel. Since the analysis is focused on the channels placed on the occipital area of the scalp, useless data can be discarded loading only data derived from the 8 channels of interest. This step is implemented by the Average Reference kernel, where each processor operates independently on the 64 samples of each time-step. Also in this case the computation on each core is completely independent, hence showing nearly ideal speed-ups with respect to the sequential execution. After Average Reference calculation, data is accumulated in the *Chunks Accumulator* (for a total of 320kB) and stored into the L2 memory.

In this application, PCA presents the most complex parallelization scheme, which is described in details in [54]. PCA is based on the Singular Vector Decomposition (SVD), to calculate eigenvalues and eigenvectors of a matrix. Both Householders and Givens matrices are used to obtain the eigenvectors via bidiagonal transformation. The former part of the algorithm cannot be parallelized at block level, because there are dependencies on the previous iterations of the bidiagonal reduction, while the latter part reaches near-ideal speedups in the parallel execution. The matrix where PCA is performed
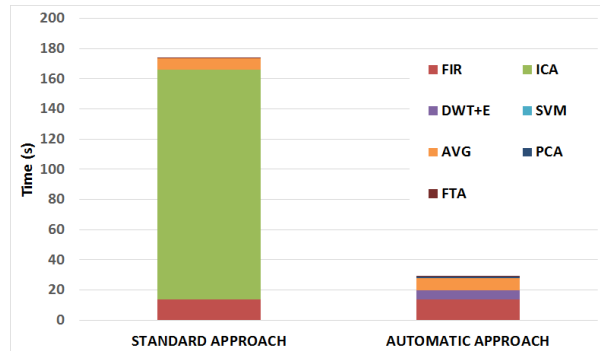

Figure 8: Estimation of execution times between Standard and Automatic Approaches.

has dimension 8x10000; the SVD part is performed on the covariance matrix of dimension 8x8, while Mean Value and Convariance Matrix and Principal Components are performed on a matrix respectively of dimension 8x10000 and 3x10000. The execution is dominated by this last two kernels achieving a total speed-up near to the ideal one. Considering the memory management, a single chunk of dimension 8x10000 is transferred to L2 memory through SPI. Then, with double buffering, smaller chunks are passed to L1 to continue the process. After PCA, 3x10000 samples are held in L2 memory (i.e. *3 Principal Components*). Then, FTA can be computed on each PC.

In the FTA stage, the most compute-intensive kernel is the FFT. The algorithm follows a butterfly diagram approach. Thus, in the parallel version butterflies are distributed among the cores and computed separately. Several synchronizations barriers are necessary at every step, so the speed-up decreases as the number of cores increases. For this reason, this kernel shows the lowest speed-up with 8 cores, but it does not affect the overall performance since FFT represent less than 1% of the entire processing chain. FFT is computed on overlapped chunks for each PCs. This data is stored in L2 and with double buffering are transferred to L1. After computing FFT, the *Power Spectrums* are summed in a vector allocated in L1 while the other data can be discarded.

4. Experimental Results

4.1. Execution Time Estimation

To provide an estimation of the computational effort required for the application, a comparison between the execution time in both the standard and the automatic approach is performed. Fig. 8 shows the execution time
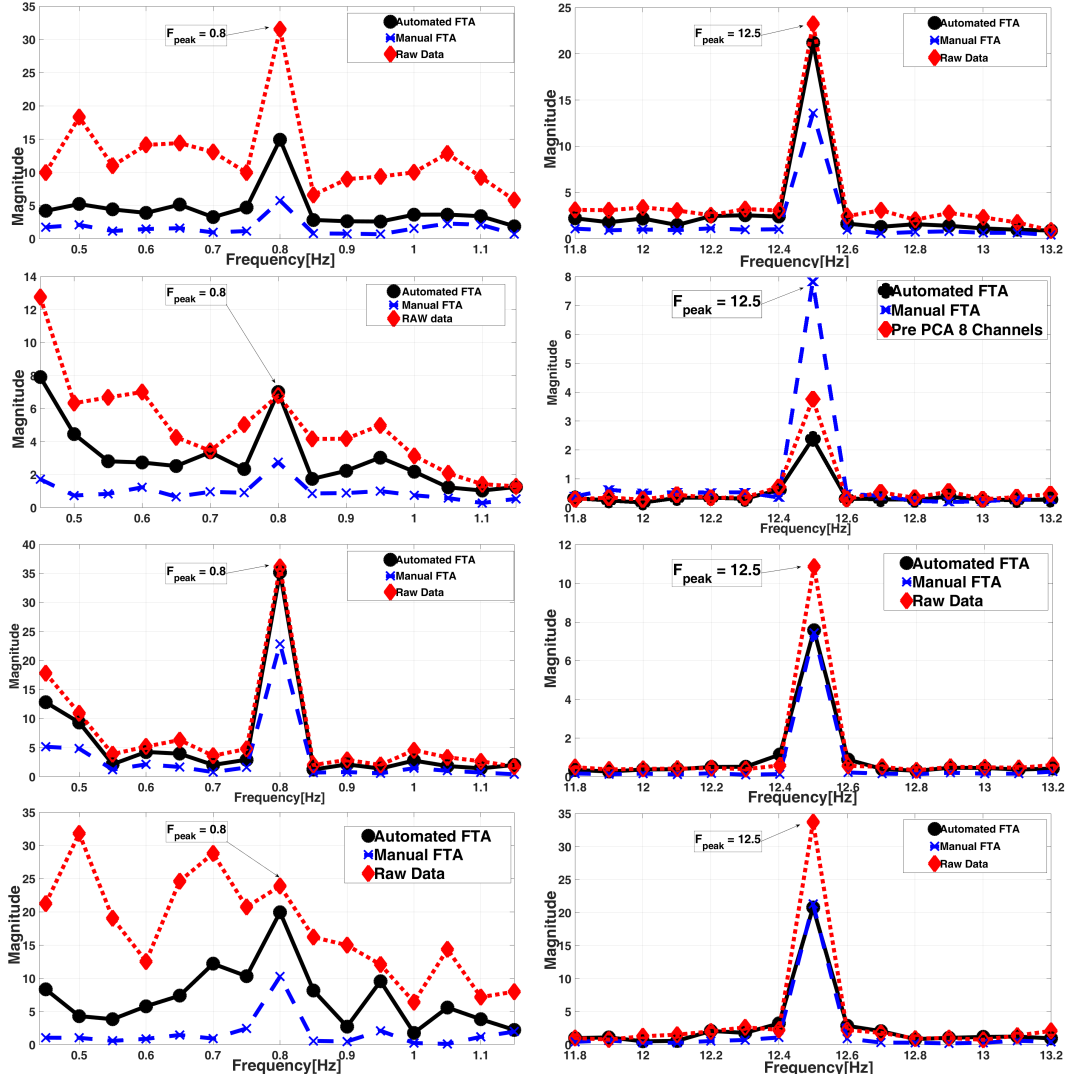
10

Figure 9: Peaks obtained from the visual stimulus at 0.8Hz and 12.5Hz. Graphs show values obtained from 4 subjects exposed to stimulus at the target frequencies.

of the two algorithms. As already mentioned, ICA decomposition is computationally intensive, hence it was excluded from the processing chain developed on the PULP platform. This choice was dictated by the necessity of finding a trade-off between accuracy and computational effort; this is mandatory during the development of embedded application, taking into account that Artifact Removal stage is enough accurate even without ICA.

The comparison is done considering the time required to execute kernels on Matlab, in seconds. As shown in Fig. 8, Automatic Approach results 6x faster than the Standard Approach. Furthermore, in Standard Approach, we are not considering the time needed to scroll and discard bad segments from the specialist, while in the Automatic Approach this task is performed on-line during the acquisition chain.

4.2. Frequency Detection

Results of the FTA are presented in Fig. 9. The power spectrum of the raw data, without sample removal, is shown in red dotted lines with diamond markers, while the power spectra obtained with the standard analysis (Section 3.2) and automated analysis (Section 3.3) are shown in blue dashed (x markers) and black solid (round markers), respectively. For convenience we present, for each subject, the data of one low stimulation frequency (0.8Hz, left-hand column) and one high stimulation frequency (12.5Hz, right-hand column). While for the high
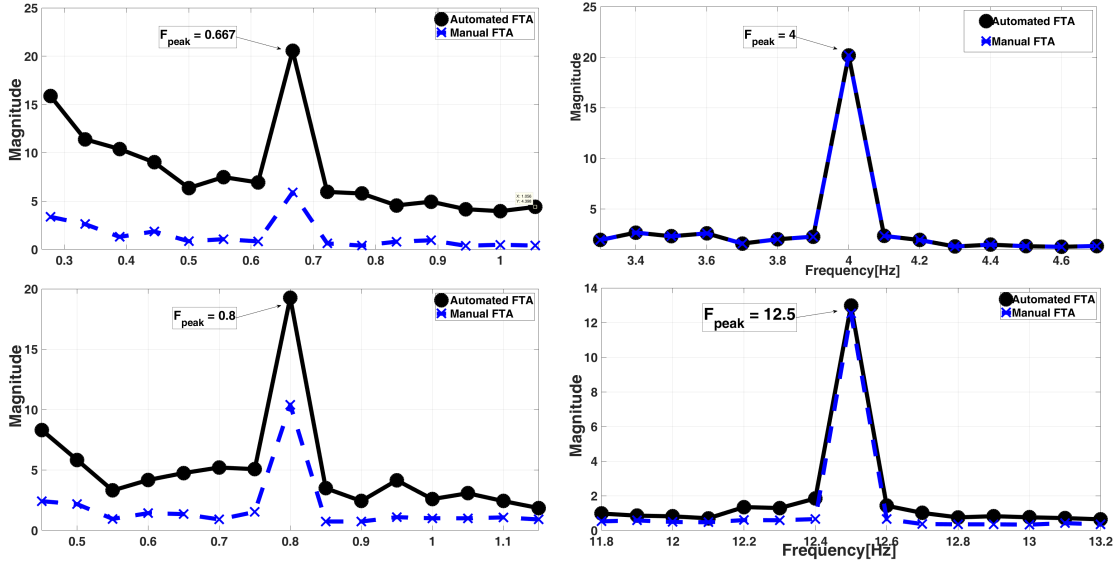
11

Figure 10: Peaks obtained from the visual stimulus at 0.667Hz, 0.8Hz, 4Hz and 12.5Hz. Graphs show mean values obtained from 4 subjects exposed to stimulus at the target frequencies.

stimulation frequency the resonant peak is evident even in the power spectrum of raw data, for the low stimulation frequency, the amplitude of the peak in the raw data is comparable to that of ongoing brain fluctuations or artifacts, therefore its identification is problematic (note in particular the raw data power spectra of subjects 2 and 4). The denoising effect of the standard analysis consists in removing a large part of unrelated fluctuations and focusing on channels containing the stronger frequency-tagged response. Therefore, even if the peak amplitude decreases, the power spectrum at neighbouring frequency bins decreases much more and flattens across the whole frequency range, resulting in a much higher normalized frequency-tagged response. The denoising effect of the automated analysis is almost as successful: the stimulation peaks are clear and detectable, even for the low stimulation frequency. An overview of the power spectra for all the stimulation frequencies averaged across the 4 subjects considered in the analysis (Fig. 10) confirms the practical feasibility of automated analysis: even though the power spectrum of the stimulus-unrelated activity is generally higher than the one obtained with the standard analysis, peaks at the stimulation frequency are also higher, resulting in a comparable peak detection efficacy, even for the low stimulation frequencies. The purpose of Frequency Detection is to understand if the brain replies to the visual stimulus properly, analyzing data acquired from the visual cortex of the subject. This means that the final response of LR helps technicians or medical staff in diagnosis of possible dysfunctions of the neurological activity (i.e. the peak is not localized or not narrowband). Thus, it can be considered as a support tool for clinicians to decide if an accurate analysis is needed to make a complete diagnosis.

4.3. Evaluation of Performance and Energy Metrics

Scaling the processing chain on a cluster of multiple cores allows to decrease the operating frequency required to achieve the real-time constrains, reducing significantly voltage supply. Thus, the quadratic dependency between supply voltage and dynamic power brings an important improvement in power density. This emphasizes a trade-off between the parallelization efficiency and voltage scaling. Execution time of the processing chain on the reference platform are shown in Table 3. Among all kernels that compose the application, FIR filter is the one that requires the majority of the computational effort. In fact, it represents the 97,88% of the execution time on PULP. As already said in the previous paragraphs, FTA kernel shows a lower speed-up compared to the others. The butterfly diagram approach used to compute FFT leads to a parallelization scheme where a high number of synchronizations barriers among cores are necessary. The weight of this barriers grows as the number of cores increases. The load of this kernel represents the 0.98% of the overall processing chain, therefore the impact on the total performance is negligible. Due to the high number of cycles required to execute the process chain, a single core

Kernel	PULP 1 core		PULP 2 cores		PULP 4 cores		PULP 8 cores	
	MCycles	load %	MCycles	Speedup[a]	MCycles	Speedup[a]	MCycles	Speedup[a]
FIR FILTER	3303,67	97,88	1651,88	1,99	826,99	3,99	413,22	7,99
DWT+ENERGY	2,50	0,07	1,25	1,99	0,63	3,98	0,32	7,83
SVM	2,57	0,07	1,29	1.99	0,65	3,94	0,33	7,69
AVG REF	0,93	0,03	0,47	1,99	0,24	3,96	0,12	7,58
PCA	32,86	0,97	16,76	1,96	8,49	3,87	4,30	7,63
FTA	38,86	0,98	21,45	1,81	13,43	2,89	9,53	4,07
TOT	3380,86	100	1693,10	1.99	850,43	3.97	427,82	7.90

[a] Speed-Up with respect to single-core PULP paltform.

Table 3: Execution of automated frequency tagging analysis on embedded computing platform

is not able to achieve the necessary operating frequency (i.e. 3.3GHz). The same constrains are worth for 2 and 4 cores (i.e. 1.6GHz and 830MHz). For these reason, a cluster of 8 cores is employed. With 8 cores an overall speed-up of 7.9x is reached, therefore, the application demonstrates to scale easily on higher number of cores. Based on these information, PULP platform can fulfill computational effort and the real-time constrains with 8 cores with an operating frequency equal to 430MHz, at the supply voltage of 0,85V consuming an average power of 55mW. As an FRAM is used as external L3 memory, an estimation of load/store operations is necessary to understand the impact on power consumption. Considering that for an access in L3 0,162nJ are required, the average power consumed is around 1,5mW. Thus, the total power consumption for the entire process chain is 56,5mW. Therefore, taking into account the power consumption and the supply voltage, approximately 66mA are required to correctly operate. Hence, assuming that this wearable device is provided with a common battery with a capacity of 2200mAh, a battery life around 24 hours can be guaranteed. For this estimation, only the major power contributors of the system are taken into account (i.e. the 8 cores in this application).

5. Conclusion

This work presented a machine learning-based, automated, embedded, and real-time EEG monitoring system targeting the analysis of the frequency tagging response in a wide range of frequency stimulations including both the standard range of SSVEPs (> 6 Hz) and the low-frequency range related to slower, higher-order neural processing (0.5-6 Hz). We compared the proposed approach with a standard method based on a mix of traditional signal processing chain (i.e. available in extraction and interpretation toolboxes such as EEGLAB) and the manual inspection of the EEG traces performed by an expert neuroscientist, on the same EEG signals from 4 subjects. We have shown that by replacing the most compute-intensive parts of the traditional processing chain with a machine learning approach we are still able to detect the presence of the same frequency peaks.

Although the standard analysis performs better in terms of signal cleaning, it is computationally intensive (i.e. because of ICA algorithm), partially manual (i.e. data scrolling and bad segments rejection) and therefore not adequate for a real-time embedded platform. Machine learning techniques are key elements for the system because they play an important role in Artifact Removal and Frequency Detection. Our proposed solution achieves 92% correct peak detection; as already shown in Section 3.3.2 this result is in line with other medical studies present in literature.

Furthermore, its implementation has been evaluated on a parallel ultra low-power platform. The overall power budget obtained is under 56mW with 8-cores, thus making the system eligible for the future wearable deployment. The next challenges to be addressed include the generalization of the machine learning models, creating a model that can fit with different subjects without incurring in loss of accuracy. An other important challenge is to identify and remove single EEG channels that are recording bad segments of data (i.e. if the electrode is broken or not perfectly stick on the scalp) without discarding good data acquired by the other channels. All these challenges will converge in reducing the complexity of the processing chain for embedded implementation while improving the accuracy of the frequency detection. This paradigm is also suitable for a wide range of low-obtrusiveness applications in healthcare and rehabilitation scenarios.

References

[1] United Nations, `http://www.un.org/` (2016).

[2] Brain Facts, `http://www.brainfacts.org/` (2016).

[3] E. Niedermeyer, F. L. da Silva, Electroencephalography: basic principles, clinical applications, and related fields, Lippincott Williams & Wilkins, 2005.

[4] D. P. Subha, P. K. Joseph, R. Acharya, C. M. Lim, Eeg signal analysis: a survey, Journal of medical systems 34 (2) (2010) 195–212.

[5] T. Thompson, T. Steffert, T. Ros, J. Leach, J. Gruzelier, Eeg applications for sport and performance, Methods 45 (4) (2008) 279–288.

[6] N. Srinivasan, Cognitive neuroscience of creativity: Eeg based approaches, Methods 42 (1) (2007) 109–116.

[7] S. J. Luck, An introduction to the event-related potential technique, MIT press, 2014.

[8] T. W. Picton, M. S. John, A. Dimitrijevic, D. Purcell, Human auditory steady-state responses: Respuestas auditivas de estado estable en humanos, International journal of audiology 42 (4) (2003) 177–219.

[9] A. M. Norcia, L. G. Appelbaum, J. M. Ales, B. R. Cottereau, B. Rossion, The steady-state visual evoked potential in vision research: a review, Journal of vision 15 (6) (2015) 4–4.

[10] G. Bin, X. Gao, Z. Yan, B. Hong, S. Gao, An online multichannel ssvep-based brain–computer interface using a canonical correlation analysis method, Journal of neural engineering 6 (4) (2009) 046002.

[11] Y. J. Kim, M. Grabowecky, K. A. Paller, K. Muthu, S. Suzuki, Attention induces synchronization-based response gain in steady-state visual evoked potentials, Nature neuroscience 10 (1) (2007) 117–125.

[12] M. Buiatti, M. Peña, G. Dehaene-Lambertz, Investigating the neural correlates of continuous speech computation with frequency-tagged neuroelectric responses, Neuroimage 44 (2) (2009) 509–519.

[13] B. Rossion, K. Torfs, C. Jacques, J. Liu-Shuang, Fast periodic presentation of natural images reveals a robust face-selective electrophysiological response in the human brain, Journal of vision 15 (1) (2015) 18–18.

[14] C. Kabdebon, M. Pena, M. Buiatti, G. Dehaene-Lambertz, Electrophysiological evidence of statistical learning of long-distance dependencies in 8-month-old preterm and full-term infants, Brain and language 148 (2015) 25–36.

[15] A. de Heering, B. Rossion, Rapid categorization of natural face images in the infant right hemisphere, Elife 4 (2015) e06564.

[16] S. Benatti, B. Milosevic, M. Tomasini, E. Farella, P. Schönle, P. Bunjaku, G. Rovere, S. Fateh, Q. Huang, L. Benini, Multiple biopotentials acquisition system for wearable applications, Proc. of SmartMedDev.

[17] S. Benatti, F. Casamassima, B. Milosevic, E. Farella, P. Schönle, S. Fateh, T. Burger, Q. Huang, L. Benini, A versatile embedded platform for emg acquisition and gesture recognition, IEEE Transactions on Biomedical Circuits and Systems 9 (5) (2015) 620–630.

[18] M. Tomasini, S. Benatti, B. Milosevic, E. Farella, L. Benini, Power line interference removal for high-quality continuous biosignal monitoring with low-power wearable devices, IEEE Sensors Journal 16 (10) (2016) 3887–3895.

[19] R. Gravina, P. Alinia, H. Ghasemzadeh, G. Fortino, Multisensor fusion in body sensor networks: State-of-the-art and research challenges, Information Fusion 35 (2017) 68–80.

[20] G. Fortino, R. Giannantonio, R. Gravina, P. Kuryloski, R. Jafari, Enabling effective programming and flexible management of efficient body sensor network applications, IEEE Transactions on Human-Machine Systems 43 (1) (2013) 115–133.

[21] G. Barbati, C. Porcaro, F. Zappasodi, P. M. Rossini, F. Tecchio, Optimization of an independent component analysis approach for artifact identification and removal in magnetoencephalographic signals, Clinical Neurophysiology 115 (5) (2004) 1220 – 1232.

[22] A. Delorme, T. Sejnowski, S. Makeig, Enhanced detection of artifacts in eeg data using higher-order statistics and independent component analysis, Neuroimage 34 (4) (2007) 1443–1449.

[23] D. Mantini, R. Franciotti, G. Romani, V. Pizzella, Improving {MEG} source localizations: An automated method for complete artifact removal based on independent component analysis, NeuroImage 40 (1) (2008) 160 – 173.

[24] Y. Li, Z. Ma, W. Lu, Y. Li, Automatic removal of the eye blink artifact from eeg using an ica-based template matching approach, Physiological measurement 27 (4) (2006) 425.

[25] F. C. Viola, J. Thorne, B. Edmonds, T. Schneider, T. Eichele, S. Debener, Semi-automatic identification of independent components representing eeg artifact, Clinical Neurophysiology 120 (5) (2009) 868 – 877.

[26] C. A. Joyce, I. F. Gorodnitsky, M. Kutas, Automatic removal of eye movement and blink artifacts from eeg data using blind component separation, Psychophysiology 41 (2) (2004) 313–325.

[27] Y. Okada, J. Jung, T. Kobayashi, An automatic identification and removal method for eye-blink artifacts in event-related magnetoencephalographic measurements, Physiological measurement 28 (12) (2007) 1523.

[28] A. Mognon, J. Jovicich, L. Bruzzone, M. Buiatti, Adjust: An automatic eeg artifact detector based on the joint use of spatial and temporal features, Psychophysiology 48 (2) (2011) 229–240.

[29] N. Ding, L. Melloni, H. Zhang, X. Tian, D. Poeppel, Cortical tracking of hierarchical linguistic structures in connected speech, Nature neuroscience 19 (1) (2016) 158–164.

[30] D. Baldauf, R. Desimone, Neural mechanisms of object-based attention, Science 344 (6182) (2014) 424–427.

[31] A. Bashashati, M. Fatourechi, R. K. Ward, G. E. Birch, A survey of signal processing algorithms in brain–computer interfaces based on electrical brain signals, Journal of Neural engineering 4 (2) (2007) R32.

[32] S. G. Mason, G. E. Birch, A general framework for brain-computer interface design, IEEE Transactions on Neural Systems and Rehabilitation Engineering 11 (1) (2003) 70–85. doi:10.1109/TNSRE.2003.810426.

[33] V. Mihajlovi, B. Grundlehner, R. Vullers, J. Penders, Wearable, wireless eeg solutions in daily life applications: What are we missing?, IEEE Journal of Biomedical and Health Informatics 19 (1) (2015) 6–21. doi:10.1109/JBHI.2014.2328317.

[34] A. Delorme, S. Makeig, Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis, Journal of neuroscience methods 134 (1) (2004) 9–21.

[35] Y. Renard, F. Lotte, G. Gibert, M. Congedo, E. Maby, V. Delannoy, O. Bertrand, A. Lécuyer, OpenViBE: An Open-Source

Software Platform to Design, Test and Use Brain-Computer Interfaces in Real and Virtual Environments, Presence: Teleoperators and Virtual Environments / Presence Teleoperators and Virtual Environments 19 (1) (2010) 35–53.

[36] R. Mahajan, C. A. Majmudar, S. Khatun, B. I. Morshed, G. M. Bidelman, Neuromonitor ambulatory eeg device: Comparative analysis and its application for cognitive load assessment, in: 2014 IEEE Healthcare Innovation Conference (HIC), 2014, pp. 133–136. doi:10.1109/HIC.2014.7038892.

[37] Quasar, http://www.quasarusa.com (2013).

[38] S. Patki, B. Grundlehner, A. Verwegen, S. Mitra, J. Xu, A. Matsumoto, R. F. Yazicioglu, J. Penders, Wireless eeg system with real time impedance monitoring and active electrodes, in: 2012 IEEE Biomedical Circuits and Systems Conference (BioCAS), 2012, pp. 108–111. doi:10.1109/BioCAS.2012.6418408.

[39] Emotiv, http://emotiv.com (2015).

[40] NeuroSky, http://www.neurosky (2016).

[41] Gtec, http://www.gtec.at (2012).

[42] OpernBCI, http://openbci.com (2014).

[43] D. Giovanelli, B. Milosevic, E. Farella, Bluetooth low energy for data streaming: Application-level analysis and recommendation, in: 2015 6th International Workshop on Advances in Sensors and Interfaces (IWASI), 2015, pp. 216–221. doi:10.1109/IWASI.2015.7184945.

[44] D. Brunelli, E. Farella, D. Giovanelli, B. Milosevic, I. Minakov, Design considerations for wireless acquisition of multichannel semg signals in prosthetic hand control, IEEE Sensors Journal 16 (23) (2016) 8338–8347. doi:10.1109/JSEN.2016.2596712.

[45] D. Rossi, A. Pullini, I. Loi, M. Gautschi, F. K. Gurkaynak, A. Bartolini, P. Flatresse, L. Benini, A 60 gops/w, -1.8 v to 0.9 v body bias ulp cluster in 28nm utbb fd-soi technology, Solid-State Electronics 117 (2016) 170 – 184.

[46] D. Rossi, A. Pullini, I. Loi, M. Gautschi, F. K. Gurkaynak, A. Teman, J. Constantin, A. Burg, I. Miro-Panades, E. Beign, F. Clermidy, F. Abouzeid, P. Flatresse, L. Benini, 193 mops/mw @ 162 mops, 0.32v to 1.15v voltage range multi-core accelerator for energy efficient parallel and sequential digital processing, in: 2016 IEEE Symposium in Low-Power and High-Speed Chips (COOL CHIPS XIX), 2016, pp. 1–3. doi:10.1109/CoolChips.2016.7503670.

[47] D. Regan, Human brain electrophysiology: evoked potentials and evoked magnetic fields in science and medicine, Elsevier, 1989.

[48] G. H. Klem, H. O. Lüders, H. Jasper, C. Elger, et al., The ten-twenty electrode system of the international federation, Electroencephalogr Clin Neurophysiol 52 (3) (1999) 3–6.

[49] B. Scholkopf, K.-K. Sung, C. J. Burges, F. Girosi, P. Niyogi, T. Poggio, V. Vapnik, Comparing support vector machines with gaussian kernels to radial basis function classifiers, IEEE transactions on Signal Processing 45 (11) (1997) 2758–2765.

[50] S. Benatti, B. Milosevic, E. Farella, E. Gruppioni, L. Benini, A prosthetic hand body area controller based on efficient pattern recognition control strategies, Sensors 17 (4) (2017) 869.

[51] J. Friedman, T. Hastie, R. Tibshirani, The elements of statistical learning, Vol. 1, Springer series in statistics Springer, Berlin, 2001.

[52] A. P. Bradley, The use of the area under the roc curve in the evaluation of machine learning algorithms, Pattern recognition 30 (7) (1997) 1145–1159.

[53] T. N. Alotaiby, S. A. Alshebeili, T. Alshawi, I. Ahmad, F. E. A. El-Samie, Eeg seizure detection and prediction algorithms: a survey, EURASIP Journal on Advances in Signal Processing 2014 (1) (2014) 1.

[54] S. Benatti, F. Montagna, D. Rossi, L. Benini, Scalable eeg seizure detection on an ultra low power multi-core architecture,

in: Biomedical Circuits and Systems Conference (BioCAS), 2016 IEEE, IEEE, 2016, pp. 86–89.

[55] D. W. Hosmer Jr, S. Lemeshow, Applied logistic regression, John Wiley & Sons, 2004.

[56] H. Chen, J. Zhang, Y. Xu, B. Chen, K. Zhang, Performance comparison of artificial neural network and logistic regression model for differentiating lung nodules on ct scans, Expert Systems with Applications 39 (13) (2012) 11503–11509.

[57] D. Delen, G. Walker, A. Kadam, Predicting breast cancer survivability: a comparison of three data mining methods, Artificial intelligence in medicine 34 (2) (2005) 113–127.

[58] B. Hosseinifard, M. H. Moradi, R. Rostami, Classifying depression patients and normal subjects using machine learning techniques and nonlinear features from eeg signal, Computer methods and programs in biomedicine 109 (3) (2013) 339–345.

[59] M. Gautschi, M. Schaffner, F. K. Grkaynak, L. Benini, 4.6 a 65nm cmos 6.4-to-29.2pj/flop@0.8v shared logarithmic floating point unit for acceleration of nonlinear function kernels in a tightly coupled processor cluster, in: 2016 IEEE International Solid-State Circuits Conference (ISSCC), 2016, pp. 82–83. doi:10.1109/ISSCC.2016.7417917.

[60] A. Marongiu, L. Benini, An openmp compiler for efficient use of distributed scratchpad memory in mpsocs, IEEE Transactions on Computers 61 (2) (2012) 222–236. doi:10.1109/TC.2010.199.

15