This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

Energy-Efficient Classification for Resource-Constrained Biomedical Applications

Mahsa Shoaran, *Member, IEEE,* Benyamin A. Haghi, Milad Taghavi,
Masoud Farivar, Azita Emami, *Senior Member, IEEE*

***Abstract*—Biomedical applications often require classifiers that are both accurate and cheap to implement. Today, deep neural networks achieve state-of-the-art accuracy in most learning tasks that involve large datasets of unstructured data. However, the application of deep learning techniques may not be beneficial in problems with limited training sets and computational resources, or under domain-specific test time constraints. Among other algorithms, ensembles of decision trees, particularly the Gradient Boosted models have recently been very successful in machine learning competitions. Here, we propose an efficient hardware architecture to implement gradient boosted trees in applications under stringent power, area, and delay constraints, such as medical devices. Specifically, we introduce the concepts of asynchronous tree operation and sequential feature extraction to achieve an unprecedented energy and area efficiency. The proposed architecture is evaluated in automated seizure detection for epilepsy, using 3074 hours of intracranial EEG data from 26 patients with 393 seizures. Average F1 scores of 99.23% and 87.86% are achieved for random and block-wise splitting of data into train/test sets, respectively, with an average detection latency of 1.1s. The proposed classifier is fabricated in a 65nm TSMC process, consuming 41.2 nJ/class in a total area of 540×1850 μm^2. This design improves the state-of-the-art by 27× reduction in energy-area-latency product. Moreover, the proposed gradient-boosting architecture offers the flexibility to accommodate variable tree counts specific to each patient, to trade the predictive accuracy with energy. This patient-specific and energy-quality scalable classifier holds great promise for low-power sensor data classification in biomedical applications.**

***Index Terms*—Gradient boosted trees, hardware architecture, on-chip classifier, decision tree, accuracy, feature extraction, latency, seizure detection, energy-quality scaling.**

I. Introduction

THE application of machine learning (ML) techniques has been exponentially growing over the past decade [1], with an increasing shift toward mobile, wearable and implantable devices. ASIC implementation of machine learning models is required to ensure a sufficiently fast response in real-time applications such as deep brain stimulation and vital sign monitoring [2]. Embedded learning at the edge and near the sensors is also critical in applications with limited communication bandwidth or privacy concerns [3]. Furthermore, to meet the tight power budget in portable or implantable devices, it is necessary to embed ML into integrated circuits rather than power-hungry FPGA-based microprocessors [4].

M. Shoaran is with the School of Electrical and Computer Engineering, Cornell University, Ithaca, 14853 NY, USA (e-mail: shoaran@cornell.edu).

B. A. Haghi, M. Taghavi, and A. Emami are with the Electrical Engineering Department, California Institute of Technology, Pasadena, 91125 CA, USA (e-mail: ballahgh@caltech.edu; mtaghavi@caltech.edu; azita@caltech.edu).

M. Farivar is with Google, Mountain View, CA 94043, USA (e-mail: masoudf@google.com).

Deep neural networks (DNNs) currently achieve state-of-the-art accuracy in most learning tasks that involve very large datasets of unstructured data (e.g., vision, audio, natural language processing). As a result, there have been significant research and development efforts to design DNN accelerators [3] and specialized ASICs, like Google's TPUs. In the context of hardware-friendly machine learning, a number of methods have been recently explored, such as reducing the bit-width precision [2], [3], sparsity-induced compression, pruning and quantization [3], and mixed-signal MAC implementation [4]. The focus of these methods is on reducing computation, data movement, and storage in neural networks.

However, application of deep learning techniques may not be practical in problems with limited computational resources, or under application-specific prediction time constraints. For instance, a common requirement of diagnostic devices is to minimize power consumption (down to microwatt-range) and battery usage, while maintaining the desired prediction accuracy and low latency. Moreover, without specialized optimization, straight-forward implementation of conventional classification techniques can be computationally intensive, requiring high processing power and large sizes of memory. Indeed, even the simple arithmetic operations performed in conventional classification methods, such as support vector machine (SVM) and k-nearest neighbor (k-NN) algorithms can become very costly with increasing number of sensors, e.g., in multichannel neural implants. Therefore, there is a need to explore alternative methods for severely resource-constrained applications.

Among other algorithms, Gradient Boosted machines, particularly the XGBoost (XGB) implementation has recently been a winning solution in ML competitions (e.g., the intracranial EEG-based seizure detection contest on Kaggle [5]). Here, we propose and optimize ensembles of decision tree classifiers and related circuit level architectures for learning applications under stringent power, area, and delay constraints, such as implantable devices. In particular, we discuss a major application of embedded classifiers in the context of closed-loop neuromodulation devices: automatic seizure detection and control in medication-resistant epilepsy. However, our techniques are broad enough to impact several other diseases and similar application domains.

With the end of Moore's Law, it is foreseeable that energy-quality (EQ) scalable systems will enable power savings that were previously provided by technology and voltage scaling [6]. EQ scaling may, in some cases, break the traditional VLSI design tradeoffs by simultaneously improving the performance, energy and area [6]. In this paper, we leverage

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

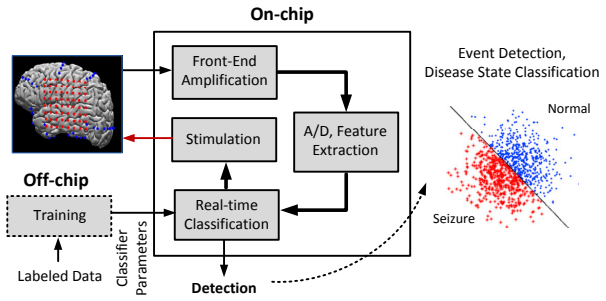


Fig. 1: General block diagram of a closed-loop system for detection and suppression of abnormal symptoms in a neurological disease. An on-chip classifier is embedded into the implantable device.

hardware-inspired techniques to implement decision tree-based classification algorithms, allowing us to employ various tree parameters as tuning knobs for accuracy, latency, and energy optimization. The resulting classifier significantly improves the power and area efficiency of conventional methods, while achieving a higher classification accuracy and sufficient latency, therefore breaking the strict energy-accuracy tradeoff. The tuning parameters include the number and depth of the trees, number of extracted features, window size, and decision update rate. By clever feature engineering and introducing an asynchronous learning scheme, a new class of scalable and low-complexity machine learning hardware for portable sensor-based applications is proposed. Specifically, we analyze the energy and quality scalability of our classifier in terms of hardware-related parameters and diagnostic performance.

This paper is organized as follows. Section II presents a review of previous methods that have been used for classification in biomedical domain, and describes their hardware cost and scalability challenges. Decision tree-based classifiers and existing hardware architectures are briefly discussed in Section III. The hardware-friendly design of XGB classifier and performance evaluation are presented in Section IV and Section V, respectively. The details of SoC implementation and measurement results are presented in Section VI, followed by a discussion on scalability and hardware optimization in Section VII. Section VIII concludes the paper.

II. Embedded Classification in Biomedical Devices

Despite major advances in medicine and drug therapy over the past decade, many disorders remain largely undertreated. Where medications are poorly effective, stimulation may offer an alternative treatment. For example, neurostimulation is today a well-established therapy for essential tremor, Parkinson's diseases and epilepsy, and has shown promise in migraine and psychiatric disorders. In particular, closed-loop neuromodulation has recently gained attention, e.g., in the form of responsive neurostimulator (RNS) for epilepsy [7], and adaptive deep brain stimulation for Parkinson's disease.

General block diagram of a closed-loop neural interface system is shown in Fig. 1. Following signal conditioning and feature extraction, an embedded classifier detects the disease-associated abnormalities in real time and triggers a programmable stimulator to suppress symptoms of the disease, e.g., a seizure or tremor, through periodic charge delivery to neurons. A high sensitivity, sufficient specificity, and low detection latency are the key requirements for the on-chip classifier, while maintaining a small footprint and low power.

Epilepsy has been one of the primary targets of neuroengineering research, along with movement disorders, stroke, and paralysis [8]. Abrupt changes in EEG biomarkers usually precede the clinical onset of seizures. Many researchers have therefore focused on extracting epileptic biomarkers for automated seizure detection [9]–[20], [21], and closed-loop control through neuromodulation [12], [14], [17].

A. Prior Work on Machine Learning SoCs

A number of classification algorithms have recently been explored for SoC implementation in diagnostic applications such as seizure detection. An 8-channel linear support vector machine EEG classifier for seizure detection is presented in [15], using the spectral energy of each EEG channel in seven frequency bins. The Gaussian basis function non-linear SVM combined with time-division multiplexing (TDM) bandpass filters in [16] achieves one of the best energy efficiencies so far (1.83 μJ/class.), a latency of 2s, and a seizure detection rate of 95.1%. Combined with front-end amplifiers and SRAM for data storage, this chip occupies an area of 25mm^2 and supports up to 8 EEG channels.

To avoid the linear growth in memory and utilized hardware with number of channels and frequency bins, a frequency-time division multiplexing approach is employed in [13], [14], along with a dual-detector classification processor utilizing two linear SVM classifiers. This closed-loop 16-channel SoC integrates a transcranial electrical stimulator, chopping amplifiers and SRAM, occupying a die area of 25mm^2. An 8-channel wireless neural prosthetic SoC is presented in [17] for intracranial EEG-based seizure control, using time-domain entropy and frequency spectrum of individual channels and linear least-square classifier. The entire system dissipates 2.8mW in a total silicon area of 13.47mm^2. A custom processor integrating a CPU with configurable accelerators for SVM classification with various kernel functions is implemented in [18]. Two medical applications including EEG-based seizure and ECG-based arrhythmia detection are demonstrated, while consuming 273μJ and 124μJ per detection, respectively. An error-adaptive boosting classifier is proposed in [19], using decision trees as weak learners. To enable controllable injection of faults, an EEG-based seizure detection system is implemented on FPGA. Dedicated accelerators combined with RISC processors are used in the 16-ch EEG-based SoCs presented in [20] and [22], implementing the fast k-NN algorithm for seizure detection, and SVM for mental status monitoring, respectively. Performance of different classifiers such as k-NN, SVM, naïve Bayes, and Logistic Regression (LR) for EEG-based seizure detection is compared in [21], where LR provides the best F1 score, area, power, and latency. A machine learning-assisted cardiac sensor SoC integrating the maximum likelihood classification (MLC) and SVM is reported in [23] for ECG-based arrhythmia detection.

It should be noted that comparison of accuracy for classifiers that are validated on different datasets or tasks, e.g., those

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

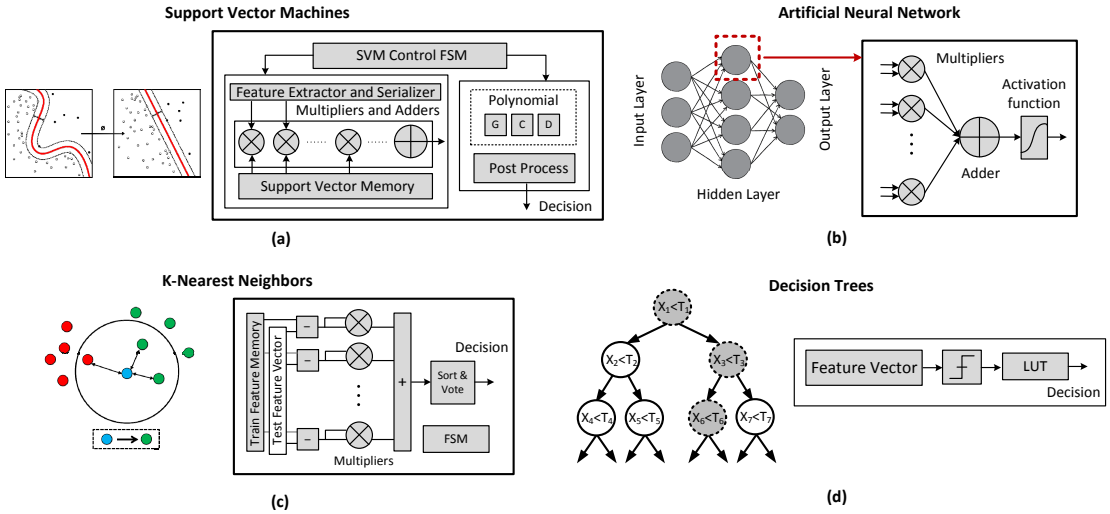


Fig. 2: Schematic of common learning models as potential candidates for hardware implementation: (a) support vector machines, (b) artificial neural networks, (c) k-nearest neighbors [21], and (d) decision tree-based classifiers.

based on EEG vs. intracranial EEG (iEEG), is not pertinent. While the main focus of our work is on hardware optimization, to evaluate the overall accuracy, we compare the proposed model to other classifiers on a large iEEG dataset [24].

In such biomedical applications, the complexity of classification algorithm, and consequently, the associated power and area, depend on the target (i.e., physician-defined) accuracy and latency for the given diagnostic task. In particular, achieving a latency of $<$2s and high accuracy with low energy consumption and small area is challenging [16]. To improve the strict energy-area-delay tradeoff and increase the number of channels, we employ a patient-specific prediction model in the form of an ensemble of decision trees, trained by the gradient-boosting algorithm. The main contribution of our work is a hardware approach that enables energy reduction by minimizing the number of simultaneously extracted features, therefore breaking the energy-area vs. accuracy tradeoff. We implement a low-complexity, yet accurate classification algorithm, that is inherently scalable to multichannel operation, through sharing the computational and memory resources among channels. In contrast to most other classifiers commonly used in literature (e.g., SVM and k-NN) that linearly scale in computational and memory requirements with number of channels and features, our proposed classifier extracts a limited number of features in a sequential fashion, regardless of total channel count. This approach enables significant savings in computational resources and storage on chip.

Moreover, we trade accuracy for lower energy, by using the most energy-efficient tree structure for a given patient and a target diagnostic accuracy. Details of our proposed method are discussed in the remainder of this paper.

Given the relative complexity of classification algorithms, the commercial devices in existence today, such as the Responsive Neurostimulator (RNS, NeuroPace) [7] for epilepsy, sacrifice the detection accuracy to meet the design constraints such as low power. The battery-powered RNS device in particular, includes three types of detectors: line length (measures the total length of the signal in a given time period), area (detects changes in signal power), and bandpass detectors. Once implanted in the skull, the selected detector by the physician is applied to a maximum of four channels and simple thresholding method is used for seizure detection. However, the detector type should be selected during the programming of device (with line-length being the default detector), which highly limits the sensitivity, specificity, and latency of seizure detection task and may result in suboptimal closed-loop control. Our proposed hardware-friendly classification algorithm would potentially improve the efficacy of current closed-loop stimulation devices such as RNS, by selective computation of features from a higher number of channels. This is achieved through a nonlinear gradient-boosting ML model that can be efficiently integrated on chip with low power.

B. Hardware Cost

When integrating a classifier on-chip, excessive memory and hardware requirements for feature extraction and machine learning, and the resulting power and area may preclude the ability to process more channels. Power consumption and chip area are mainly determined by the type and number of features, the number of channels monitored, and the type of classifier. The hardware costs associated with feature computation and classification tasks are discussed below.

1) Feature Computation Complexity: Various characteristic features can be extracted from neural data to detect the onset of a particular disease state. A major drawback of common classification methods, with the exception of decision trees, is that they must extract all required features from every input channel to classify the data. Therefore, they require extensive computational resources. Filter banks that are commonly used for spectral power extraction in non-overlapping bands are a key to diagnose neurological disorders and many other signal classification problems, e.g., voice detection, sleep-state classification, irregular heartbeat detection. For instance, to implement the SVM classifier in [18], the band-limited components in eight different bins are extracted from EEG, using

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

FIR filters. The energy of each component is accumulated in a 2s window, and the features from three consecutive windows are combined, resulting in a feature vector with a dimension of 8×3×N, where N corresponds to number of EEG channels. However, filters are computationally intensive due to MAC operations. Various methods have therefore been explored to reduce the number of multiplications needed or the associated overhead, such as matrix-multiplying ADCs [25], TDM [16], and frequency-time division multiplexing [14].

In contrast to low-frequency EEG-based systems [9], [10], [16], [18], at higher frequencies associated with iEEG where high-frequency oscillations (HFOs) are among relevant biomarkers [26], a larger number of bandpass filters is necessary. Moreover, depending on the application, the use of complex and non-linear features may be inevitable. Selecting a small subset of hardware-friendly features [7], [21], [11] can help to meet the power and area constraints, but may sacrifice the classification accuracy. These classifiers also require combinations of serializers, MUX/DEMUX circuits, and buffers to store and process input data and features.

2) Classification Complexity: Simplified schematic of some of the common classifiers for sensor data classification are shown in Fig. 2. Neural Networks (NNs) are hardware intensive and typically require high processing power to perform complex computations, as well as large amounts of memory to store many parameters on chip. Furthermore, due to limited access to training sets and patient-specific biomarkers in biomedical applications such as seizure detection (that require extensive monitoring in an invasive setup at the hospital), NN and Deep Learning classifiers would generally result in a poor classification accuracy.

SVM with its intrinsic characteristics such as easy modeling, reproducible results, and robustness through convergence to global minima, has been the most commonly used classifier for epileptic seizure detection from EEG [14]. Three SVM kernels have been applied to on-chip seizure classification: linear, second-order polynomial, and Gaussian SVM (RBF) [14]. The latter achieves better tradeoffs between classification accuracy and latency, with more complex implementation. However, both polynomial and Gaussian SVM require sufficient seizure patterns for training to achieve high accuracy, which is not the case for patients with limited seizure data available [14]. The general classification function of SVM is given by:

$$f(x) = \sum_{i=1}^{N_{sv}} \alpha_i K(s\vec{v}_i, \vec{x}) + b \qquad (1)$$

where $\vec{x}$ is the feature vector, $s\vec{v}_i$ is one of the N_{sv} support vectors, K is a kernel function, α and b are the modeling parameters. Even though SVM has demonstrated impressive performance in seizure detection from EEG [16], [15], [9], [18], the computational complexity of the decision function in (1) depends on the type of kernel [27]. Generally, a large number of support vectors is required to yield high accuracy in seizure detection, and using a strong classification kernel such as RBF, the energy scales proportionally, dominating by orders of magnitude over feature extraction, front-end, and digitization [18]. While the primary computations for polynomial and linear kernels are dot-product and weighted summation over support vectors, the RBF kernel requires subtract-square accumulation, exponentiation (commonly implemented via CORDIC), and weighted summation over the support vectors [18]. Excluding the nonlinear kernel, the hardware complexity (i.e., number of multiplications and additions) is proportional to $N_{sv} \times d$, where N_{sv} is the number of support vectors and d is the dimensionality of the feature vector [27]. The number of required support vectors depends on separability of the features. A greater number of support vectors is needed for highly nonlinear separation boundary between classes. While more computational resources are available in EEG monitoring systems, the high computational complexity of the RBF kernel makes it unsuitable for implementing in an implantable device that acquires iEEG signals from within the brain (similar to RNS device [7]). The linear SVM would reduce the complexity of the seizure detection algorithm. However, the performance may be degraded if the features are not linearly separable [27].

k-NN classification requires computing the distances between the test and training features, while tracking the k smallest distances. While showing a good performance for epileptic seizure detection [20], the large size of the training set memory and the exhaustive search for nearest neighbors make the classifier power demanding [20]. Moreover, k-NN is more suitable for classification tasks with large sample sizes. In [21], the k-NN classifier achieves a higher F1 measure in seizure detection than the linear SVM, but it consumes dramatically more FPGA resources and power [21].

A simple NN has inputs being multiplied by a weight vector, added together and followed by a linear or nonlinear function to generate the output to the next stage. Logistic regression (similar to a one-layer neural network) uses a linear weighted combination of features and generates the probability of different classes. In general, such methods may not be well suited for efficient hardware implementation due to the complexity involved in feature extraction and classification.

Individual decision trees (DTs) and their ensembles, such as Random Forests and Gradient Boosting, are among the most useful and highly competitive methods in ML, particularly in the regime of limited training data, little training time and little expertise for parameter tuning. Authors in [27] propose a non-linear classifier using AdaBoost technique with decision stumps (trees of depth one) as base classifier, to enable a low-complexity seizure detection system. The relative hardware efficiency of DTs is evident from the fact that simple digital comparators form the main processing unit of a DT, with no need for multiplications, as illustrated in Fig. 2(d). In [28], AdaBoost performs slightly better than SVM with less hardware complexity, achieving a sensitivity of 77.1% (tested on 873h of iEEG data) and a false alarm rate of 0.18/hour. The hardware complexity of AdaBoost depends on the required numbers of comparison operations, which is equal to the number of decision stumps (60 in [28], with average feature set size of 14.6). Given their reduced training complexity, DTs are chosen among the various classifiers that have been considered for boosting (e.g., SVMs, NNs) to implement the error-adaptive classifier proposed in [19].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

A detailed discussion on hardware implementation of DTs is presented in Section III. Given the variety of hardware schemes used for different arithmetic units in classification and feature extraction, we opted to use a unified metric for evaluating the overall computational complexity of our design and comparing it to prior works, by reporting the number of equivalent 2-input NAND gates. This measure is provided in the SoC comparison table in Section VI.

C. Scalability Challenges in Multi-Sensor Systems

Several studies show that a large number of acquisition channels are required to obtain an accurate representation of brain activity for disease diagnosis or movement decoding, and the therapeutic potential of neural devices is limited at low spatiotemporal resolution [29]. Similar concerns apply to cardiac implants and ECG electrode arrays. Therefore, it is expected that future interfaces integrate hundreds of channels, posing extreme constraints on power dissipation of the circuits. Besides, efficient realization of wearables and IoT devices requires integration of multi-sensor platforms with embedded machine learning techniques and real-time analytics.

Despite substantial research on machine learning, hardware-friendly and scalable implementation is not sufficiently addressed. Even the simple arithmetic operations performed in conventional classification methods can become very costly with increasing number of channels and feature dimensions. For instance, the size of feature vector $\vec{x}$ in (1) linearly increases with number of channels, and so does the number of multiplications and additions required in a linear SVM. Furthermore, the current method of extracting features separately from each channel requires either a dedicated ADC and feature extraction unit per channel, or power-hungry multiplexing circuits and buffers. Extensive system-level optimizations, specialized hardware techniques, and new design paradigms are needed to meet the energy and accuracy requirements, while preserving the high-channel-count recording capability, that has been addressed in this paper.

III. Decision Tree-based Classifiers

Decision tree (DT) [30] is a popular non-linear ML model where the target class is determined by a sequence of queries, i.e., comparison to a threshold, on input features that start at the root node and terminate in a leaf node, as shown in Fig. 2(d). Compared to NNs, tree-based classifiers are extremely fast in training and classification, and require far fewer parameters for tuning. They can be easily parallelized, and are robust to label noise. With simple comparators as their building blocks, DTs are naturally a viable solution to reduce complexity [31]. However, the conventional hardware for DTs may not provide optimal results.

In [32], a wearable gait monitor using DTs achieved roughly identical detection accuracy to SVMs, drawing 3× less power. While DTs are commonly implemented in software, there are a few works that implement DTs in hardware. A decision tree spike sorting classifier was reported in [33]. The feature at the active node is multiplexed from a total of four features extracted from the spikes in a neural channel. Authors in [34] present an acoustic front-end for speech classification using decision trees. A set of potential features (e.g., band-powers using 8 analog bandpass filters in parallel) are extracted from the input signal, and the feature at each node is multiplexed from this set. The decisions are made by logically combining the outputs of all nodes in a tree, e.g., 7 nodes in Fig. 2(d).

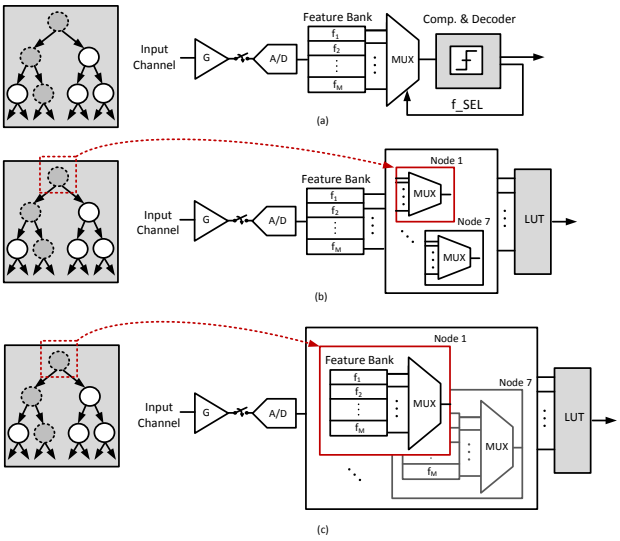


Fig. 3: Block diagram of conventional DT architectures for a single input channel.

A. Conventional Hardware Architectures

Although the hardware solutions presented in [33], [34] are suitable for applications with limited number of features and scarce activity (e.g., spike sorting/voice detection where the classifier and feature extractor are only active when a spike/voice is detected), or limited input sources (e.g., voice detection), extending this approach to multi-sensor systems with more features is challenging and can be power-hungry.

As illustrated in Fig. 3, the direct implementation of DTs requires initial extraction of all features from the input data [33], [34], Fig. 3(a), (b), or allocation of a separate feature extraction unit to each node, Fig. 3(c). In problems dealing with multichannel and multi-feature signals, particularly where a combination of trees is required to obtain a higher accuracy, the utilized hardware by each tree must be minimized. For example, assuming a 100-channel neural recording array and a set of 10 features per channel (typical for seizure detection), the first two architectures would require initial processing of a thousand features, the associated memory, and multiplexing circuits. Yet, only a small portion of these features are employed in the classification task, that is the sum of visited nodes in all trees ($\leq$ maximum depth $\times$ number of trees). Similarly, the third method would require 7 feature extraction and multiplexing units per tree, as depicted in Fig. 3(c). Since a maximum of one node at each level of the tree is visited, we previously proposed to utilize one feature extraction unit per level [31], to reduce the required hardware resources compared to Fig. 3(c).

To support multichannel operation, the alternative approach of placing a tree per channel would require the allocation of a separate DT hardware to each channel. However, in case of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

disease detection, it is likely that only a small subset of channels capture the abnormal activity, e.g., the electrodes placed in seizure foci. Therefore, training a classifier on the entire array rather than separately classifying every single channel would avoid the unnecessary extraction of features from silent channels. In summary, while DTs offer significant advantages to other classifiers by avoiding multiplication and using fewer memory units, the existing hardware is not well-suited for high-channel-count and resource-limited applications.

IV. Hardware-Friendly XGB Classifier Design

Here, we propose a hardware-efficient online classification algorithm using an ensemble of gradient-boosted decision trees, as illustrated in Fig. 4. Essentially during a classification task by a decision tree, only one path from the root to the leaf is visited. Therefore, unlike other classifiers, only a limited number of features are necessary in practice to make a decision. These features, however, are carefully selected by employing powerful training algorithms that produce the optimal tree structure to maximize the overall predictive accuracy. The trained prediction model, which is the output from the gradient-boosting algorithm, includes full information on tree structures in the ensemble such as thresholds, leaf values, and selected features (shown as Serial Control IN in Fig. 4, where CH_i and FC_i represent the channel number in the array and feature number, respectively).

The intuition behind our hardware architecture is the following. Since the decision of each tree is made upon completing a series of successive comparisons, a single feature extraction module (and the preceding ADC) can be sequentially used to exclusively calculate the requested feature at the current node. The split direction and next active node are determined by comparing this feature with the corresponding threshold. Therefore at each step, only the selected channel is used for online feature extraction, without buffering the data from other channels or extracting unnecessary features. As shown in Fig. 4, the final answer is the sum of answers of all trees (details are discussed below).

In our proposed architecture (Fig. 4), an ensemble of up to eight gradient-boosted decision trees, each with a fully programmable Feature Extraction Engine (FEE) including FIR filters continuously process the input channels. In a closed-loop architecture, the FEE reuses a single filter structure to execute the top-down flow of the decision tree, where FIR filter coefficients are multiplexed from a shared memory. This approach results in significant hardware saving, compared to the methods shown in Fig. 3. A potential drawback of this serial processing approach would be the degraded latency, that is carefully studied in this Section.

A comparison of hardware complexity for various DT architectures (assuming a single tree) is summarized in Table I, where N, M, and l represent the channel count, maximum number of nodes, and depth of a tree. The proposed architecture enables the lowest number of FEEs and classification hardware, and therefore, the lowest complexity. The number of FEE modules (or number of computed features) linearly increase with number of channels in the first two methods. Although our proposed architecture reduces the number of

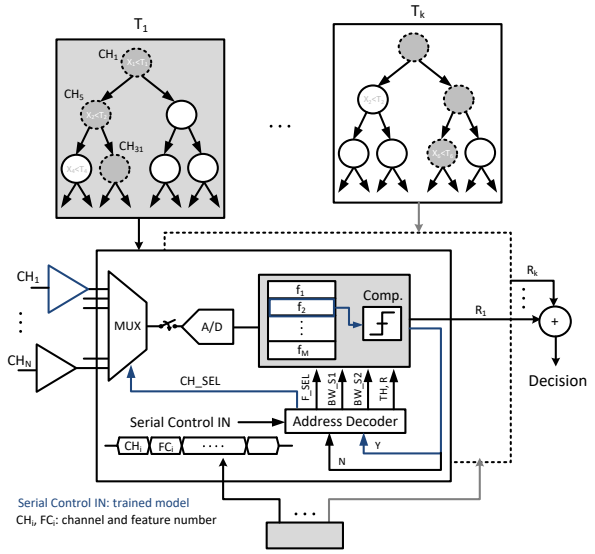


Fig. 4: Proposed hardware architecture for an ensemble of gradient boosted decision trees.

TABLE I: Hardware Complexity of DT Architectures

Architecture	# of FEE	# of Comparator	# of MUX
Fig. 3(a)	N	1	1
Fig. 3(b)	N	M^*	M
Fig. 3(c)	M	M^*	M
[31]	l	l	l
This work (XGB-HW)	1	1	1

*Additional LUT is needed to generate the final decision.

feature extraction and classification (i.e., comparator and multiplexer) units, the memory needed to store the tree structure and coefficient values remains the same in all architectures in Table I. The detailed memory breakdown of our proposed scheme is further discussed in this paper.

A. Gradient Boosted Trees

Gradient-boosting [35] is one of the most successful machine learning techniques that exploits gradient-based optimization and boosting, by adaptively combining many simple models to get an improved predictive performance. Binary split DTs are commonly used as the "weak" learners. Boosted trees are at the core of state-of-the-art solutions in a variety of learning domains, given their excellent accuracy and fast operation. For example, among the 29 challenge winning solutions published on Kaggle in 2015, 17 used XGB, where DNN was the second most popular method, used in 11 solutions [36].

Boosting involves creating a number of hypotheses $h_t(x)$ and combining them to form a more accurate composite hypothesis. The output of a boosted classifier (or regressor) with an input feature vector of x has the additive form of

$$H(x) = \sum_t \alpha_t h_t(x). \tag{2}$$

where α_t indicates the extent of weight that should be given to $h_t(x)$. A general schematic diagram illustrating an ensemble of depth-3 trees is shown in Fig. 5. Using gradient-boosting, the trees are built in a greedy fashion to minimize a regularized objective on the training loss [36].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

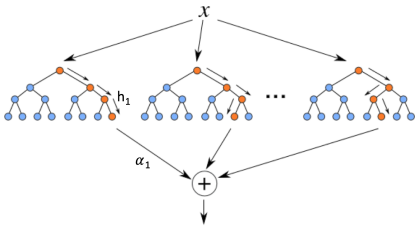


Fig. 5: Schematic diagram of a boosted ensemble of decision trees.

In this paper, we have employed the XGBoost package [36], a parallelized implementation of the gradient boosting algorithm. To assess the performance of proposed classifier on a relatively large dataset, epilepsy is chosen as our case study, given the availability of continuous recordings from many patients. This architecture, however, can potentially benefit many other on-chip sensor signal classification problems. Applying XGB to our iEEG dataset, we observed over 100 times improvement in training speed compared to common SVM implementations.

In the proposed hardware (Fig. 4), given that only one channel is used at each feature computation step in a tree, the rest of input channels can be switched off to save power. For example, to classify a 100-channel neural data with 8 trees, only 8 channels are simultaneously active. In contrast to SVM and other methods that require all features from the entire array, this approach significantly reduces the memory and hardware overhead. To reduce energy, a minimum number of trees that obtain a sufficient accuracy are used, that is chosen upon training. Moreover, as a significant advantage, only one tunable bandpass filter can be used to extract as many band-power features as needed, since these features are not computed in parallel. By employing a programmable FIR (or tunable analog) filter, the corresponding coefficients (or band selection parameters) can be easily multiplexed from memory, according to the feature being processed, as shown in Fig. 4. Besides, as shown later in this paper, very little improvement in performance is achieved by using trees with a depth of 4 and above. Therefore, these ensembles can be made by a relatively small number of low-depth trees, resulting in significantly lower computational complexity than conventional models, as later confirmed in our comparison table in Section VI.

B. Delay Constraint

The proposed architecture faces a practical challenge of designing decision trees under application-specific delay constraints. Given any ensemble $\mathcal{T} = \{\mathcal{T}_1, \ldots, \mathcal{T}_k\}$ of decision trees obtained from our original method, we need to ensure that each tree $\mathcal{T}_i$ satisfies the delay constraint:

$$\sum_{i \in \pi(h)} d_i \leq \Delta T \tag{3}$$

where d_i is the time required to compute feature f_i, ΔT is the maximum tolerable detection delay, and $\pi(h)$ is the set of all predecessors of node h. One possibility is using a "greedy" algorithm to solve this practical constraint by building trees that satisfy the delay requirement, as depicted in Fig. 6. However, this algorithm may result in a suboptimal solution,

Input: Original trained tree ensemble $\mathcal{T} = \{\mathcal{T}_1, \ldots, \mathcal{T}_k\}$
Output: Delay-constrained ensemble $\mathcal{T}' = \{\mathcal{T}'_1, \ldots, \mathcal{T}'_k\}$
Data: training set: $S = \{(x_i, y_i)\}$
feature set: $F = \{f_i\}$, each with delay d_i
delay tolerance: ΔT
set of predecessors of node h: $\pi(h)$
for *all trees* $\mathcal{T}_i$ *in* $\mathcal{T}$ **do**
  for *each node* $h \in \{1, \ldots, |\mathcal{T}_i|\}$ **do**
    if $\sum_{i \in \pi(h)} d_i > \Delta T$ **then**
      $\forall f_i \in F$ find feasible f that obtains the best $SplitCriterion(f_i, S)$
      Label node h with f
      Grow $Subtree(h)$
    end
  end
end

Fig. 6: A greedy training algorithm to meet the delay constraint.

since the split criterion and subsequent feature selection is subject to the hard constraint on delay.

C. Asynchronous Tree Operation

To solve this issue, we introduce an asynchronous approach where trees freely run in parallel, each with features that maximize the accuracy, regardless of their computational delay. Using the averaged results of completed trees and previous results of incomplete trees, decisions are frequently updated to avoid long latencies.

1) Decision-Making Procedure: First, we need to select an optimum time to update the decision of the system. Suppose that we have k trees represented by $\mathcal{T}_i$, $i \in \{1, 2, \cdots, k\}$. Assuming that t_i is the total time associated with the longest path in $\mathcal{T}_i$, we select the optimum update time as:

$$t_{opt} = \min\{t_1, t_2, \cdots, t_k\} \tag{4}$$

This guarantees that at least one tree will be completed in this interval, and a new decision is made every t_{opt}. Then, we calculate the average value of decisions for each tree:

$$D_{\mathcal{T}_i} = \frac{1}{N_i} \sum_{j=1}^{N_i} r_j \tag{5}$$

where N_i is the number of completed cycles over t_{opt} and $r_1, r_2, \cdots, r_{N_i}$ are the corresponding results (i.e., leaf values) of $\mathcal{T}_i$. In a boosting classifier, the answers of all trees must be summed up to make the final decision. Positive answers are classified as seizure and negative ones as non-seizure. The final result of the system is therefore updated as below:

$$D_{final} = \sum_{i=1}^{k} D_{\mathcal{T}_i} \tag{6}$$

In case there is no new answer for tree $\mathcal{T}_i$ after t_{opt}, we simply use its previous decision. By employing this approach and assuming an initial setup time, there always happens to be at least one result produced during t_{opt} to make a decision.

In the proposed asynchronous architecture, each tree continues to test the input data, without waiting for other trees to complete. Suppose that x is a test input that moves through the tree. As x enters node i, it takes time d_i to calculate

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

TABLE II: Patient Data and Signal Acquisition Info.

Subj.	iEEG Portal ID	No. Elec.	No. Seiz.	Rec. Dur.	Samp. Rate
1	Study 004-2	56	3	7d 18h	500
2	Study 006	56	5	1d 14h	500
3	Study 017	16	9	7d 17h	500
4	Study 011	88	3	3d 12h	500
5	Study 022	56	7	3d 23h	500
6	Study 023	88	4	2d 5h	500
7	Study 012-1	60	6	3d 7h	500
8	Study 027	48	6	3d 21h	500
9	Study 016	64	7	5d 21h	500
10	Study 031	116	5	6d 19h	500
11	Study 030	64	8	5d 23h	500
12	Study 020	56	8	5d 0h	500
13	Study 014	104	15	6d 0h	500
14	Study 021	108	13	6d 11h	500
15	Study 026	96	22	3d 3h	500
16	Study 024	88	19	8d 10h	500
17	Study 028	96	9	1d 16h	500
18	Study 038	88	10	3d 0h	500
19	Study 005	16	151	6d 16h	500
20	I001_P034_D01	47	16	1d 8h	5k
21	Study 040	116	6	2d 23h	5k
22	Study 036	96	4	4d 14h	5k
23	Study 019	96	36	5d 16h	500
24	Study 033	128	17	6d 17h	500
25	Study 029	64	3	5d 1h	500
26	Study 037	80	8	8d 23h	500

TABLE III: Evaluated Features

Feature	Description
Line-Length (LLN)	$\frac{1}{d}\sum_d \lvert x[n]-x[n-1]\rvert$, d = window length
Power (POW)	Total spectral power
Variance (VAR)	$\frac{1}{d}\sum_d (x[n]-\mu)^2$ where $\mu=\frac{1}{d}\sum_d(x[n])$
Delta (δ)	Spectral power in 1-4Hz
Theta (θ)	Spectral power in 4-8Hz
Alpha (α)	Spectral power in 8-13Hz
Beta (β)	Spectral power in 13-30Hz
Low-Gamma (γ_1)	Spectral power in 30-50Hz
Gamma (γ_2)	Spectral power in 50-80Hz
High-Gamma (γ_3)	Spectral power in 80-150Hz
Ripple	Spectral power in 150-250Hz
Fast Ripple (FR)	Spectral power in 250-600Hz (@ SR = 5kHz)

the feature f_i. Based on the value of f_i, a split to either right or left branch is made, and the process continues until a leaf is reached. By effectively averaging the decisions of fast trees over multiple cycles, while allowing the longer trees to complete, we show that the overall performance of this online asynchronous approach is even superior to the conventional offline method [31], where features at different nodes are simultaneously extracted over the same window and decisions are made at the end of this window (a hardware-intensive solution). Since it is likely that more than one answer would be provided by t_{opt}, averaging can reduce the impact of noisy decisions. Moreover, features are extracted from successive parts of the decision window, rather than one feature for the entire window. Therefore, the decisions are more accurate, while the optimum selection of update time in (3) reduces the detection latency.

V. Performance Evaluation

As a benchmark, we consider a boosted ensemble of 8 trees with a maximum depth of 4 using proposed model (XGB-HW), and compare it to the linear, cubic, and RBF SVM, k-NN with 3 and 5 neighbors, Logistic Regression, offline XGB (abbreviated as XGB) [31], Random Forest and Extra Tree classifiers, both with 8 trees and a maximum depth of 4. A hyperparameter tuning of classifier parameters was performed to find optimum settings.

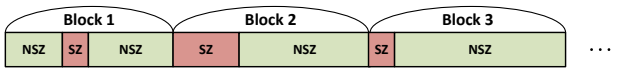


Fig. 7: The proposed block-wise data partitioning, where SZ and NSZ represent the seizure and non-seizure segments, respectively.

A. Data Description

In this work, we use the publicly available data from the intracranial EEG portal [24]. Continuous recordings from 26 patients sampled at either 500Hz or 5kHz are included in our study. The seizure events are marked by physicians, and patients have been recorded at varying channel counts (ranging from 16 to 128). The access IDs of analyzed patients and further details are provided in Table II. Overall, we studied a total of 3074 hours of iEEG including 393 seizures.

B. Train/Test Split

A common problem in performance evaluation of real-time classifiers such as seizure detectors is to randomly partition the entire data into train and test samples. Shuffling provides prior information from parts of test data (that should remain unseen) during training, resulting in data leakage. We use a block-wise splitting approach to avoid this problem and fairly assess the performance of our classifier for practical test conditions such as seizure detection. In the block-wise method shown in Fig.7, we divide the continuous iEEG data into seizure and non-seizure segments, where each seizure is concatenated with the following non-seizure segment into a larger "block" (the first non-seizure segment is added to the beginning of first block). Thus, each block is comprised of a complete seizure attached to the following non-seizure segment. Most patients in our dataset have sufficient and long enough seizure data to allow this approach. However, cases with small number of short seizures are not good candidates for block-wise selection. Therefore, we removed two patients from our initial dataset.

For the purpose of feature extraction during training and offline testing, we divide the time series into 1s windows and extract all features from channels for each window. We compare our block-wise method with the commonly used random split, in which a 5-fold cross-validation is applied to the shuffled data, followed by a hyperparameter tuning to maximize the F1 score for all classifiers. To tune the parameters for the block-wise approach, we apply a block-wise 5-fold cross validation. In this case, 20% of blocks (rounded up to the nearest integer) are retained for testing the model, and the remaining are used as training set. The cross-validation process is then repeated for 5 times and the results are averaged to produce a single estimation. For patients with less than 5 seizures, we opted for a block-wise leave-one-out approach, where we use one block as test and the remaining blocks as train, and repeat this for all blocks. To evaluate the corresponding F1 score, sensitivity, and specificity, we use the tuned parameters for each patient and average the results of cross validation tests as described above. For XGB-HW, the trained prediction model generated by the gradient-boosting algorithm includes all the information on tree structures such as leaf values, thresholds and selected features. Using this

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

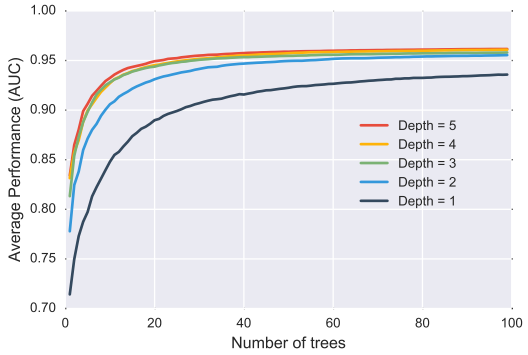
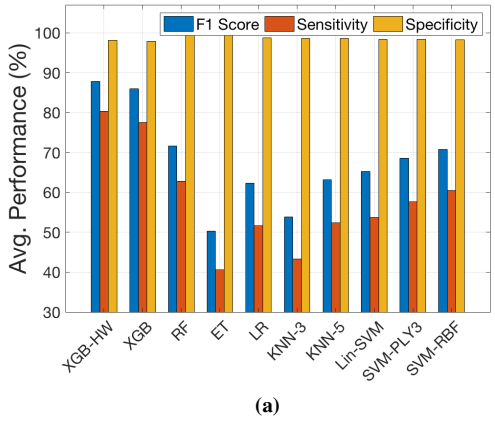
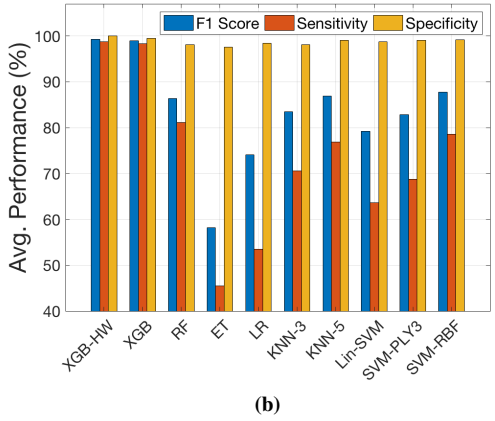


Fig. 8: The overall classification performance at various depths versus number of trees.



(a)



(b)

Fig. 9: Comparison of average predictive ability (F1 score), sensitivity, and specificity of different classification methods among patients, using (a) block-wise, and (b) random splitting methods, respectively.

trained model, the online XGB classifier is tested according to the procedure described in Section IV. C. To minimize the update interval and latency, features are extracted over smaller time windows than 1s.

C. Feature Extraction

Prior works [37]-[41] have extensively analyzed the optimal features for seizure onset detection. For instance, line-length achieves the best seizure detection performance among more than 65 different time and frequency-domain features in [37]. This time-domain feature is a measure of line length between successive samples and provides an appropriate characteristic of epileptiform iEEG, since it increases at both low-amplitude fast and high-amplitude slow activities, that normally occur prior to a seizure [39]. Another frequently used feature is the energy of the signal, as a measure of signal power over time. It was firstly shown in [38] and later by several investigators [39]-[41] that the power and variance of EEG/iEEG signals are increased minutes prior to seizure onset. In addition, many studies on EEG signals have been focused on spectral power features in the range of below 30Hz (i.e., the Berger bands) [37], [9], [15]. However, the iEEG signals span a wider frequency range and go beyond 200Hz for seizure biomarker extraction [26]. These high-frequency oscillations (HFOs) have been previously studied by many researchers [26], [42]. The authors of [42] have concluded a significant potential of HFOs for seizure detection from iEEG.

Based on our initial study on discriminative performance of several frequency and time domain features [31], and the existing literature [37]-[41], we chose the following set of features: line-length, total power, time-domain variance, and power in multiple frequency bands, as listed in Table III. We previously analyzed the discriminative performance of this feature set on an extensive iEEG database [31], in which line-length was the best discriminative feature. While the optimal frequency range was patient-dependent, in majority of patients sampled at a sufficiently high rate (5k), it had a clear shift from low-frequency bands toward gamma, ripple, and fast ripples.

Rather than using the absolute value of spectral power [31], normalized features were calculated by dividing the spectral power within each frequency band by the total power. The power values (and corresponding thresholds) typically change with the daily life status of a patient, such as sleep state, physical or mental activities, and consciousness level [43]. In contrast, normalized values are more robust with respect to fluctuations in a patient's daily life and have been utilized in our study. Features are obtained from each iEEG channel using 1s windows for training and offline testing. During online testing, we assign a minimum extraction time to each feature, based on their computational delay. Using normalized band powers, we observed an improved seizure detection accuracy compared to absolute spectral power features used in [31].

It should be noted that various other features may be included to enable more accurate seizure detection. However, the focus of this work is on the classification algorithm. The literature pertaining to analysis of various features for epilepsy diagnosis is immense, and can be found in [37]-[41].

D. Depth and Number of Trees

Decision trees are very efficient, but also susceptible to overfitting in problems with high feature-space dimensionality. To address this, we limit the number of nodes in each tree, i.e., design shallow trees using small number of features [31]. Shorter trees are also more efficient in hardware and incur less detection delay. Figure. 8 shows the area under the curve (AUC) performance of an ensemble of gradient-boosted trees versus the number of trees for different values of depth parameter. An important observation is that the detection

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

accuracy is not significantly improved ($< 0.5\%$) with depth values of 4 and higher. Besides, an AUC higher than 90% is achieved using fewer than 10 trees of depth 3 or 4. Therefore, the total energy can be minimized by limiting the number of trees and depth, which are chosen as 8 and 4 in our study.

E. Performance and Comparison

The average performance of classifiers across patients are shown in Fig. 9(a) and (b), using block-wise and random splitting methods, respectively. As mentioned before, due to correlation of iEEG waveforms, random splitting can allow the model to learn from parts of test data and statistics of unseen seizures during training. Therefore, it creates overly optimistic predictive models and invalidates the estimated performance. In this paper, we consider block-wise approach to alleviate the leakage problem. The F1 score is calculated by counting the number of correctly classified windows, given by:

$$F_1 = \frac{2}{\frac{1}{Sen.} + \frac{1}{Spec.}} \tag{7}$$

where sensitivity and specificity represent the true positive and true negative rates, respectively. The asynchronous XGB (XGB-HW) performs best among all classifiers, reaching an average F1 score of 99.23% and 87.86%, for the random and block-wise splitting methods, respectively, with an average block-wise sensitivity of 80.33% and specificity of 98.12%. This is achieved by efficient design of the learning algorithm in an asynchronous online fashion, while minimizing the hardware resources and energy. As expected, random split leads to higher, but unrealistic predictive accuracy. Interestingly, only tree-based methods, in particular, the XGB could classify patient 21's seizures (87% F1 score), while all other classifiers failed for this patient. Random forests generally require a large number of trees to obtain a high performance, which is not suitable for on-chip implementation. Our results indicate that the proposed asynchronous gradient-boosting method with as low as eight trees, has a higher generalization ability on this iEEG dataset, compared to methods such as k-NN, LR, and SVM. The performance could be further boosted by artifact removal, as some datasets (e.g., patient 13) are contaminated by high-frequency artifacts that particularly overlap with FR band. To evaluate the detection latency, we count the number of correctly classified ictal windows at the beginning of a seizure, and wait for at least three consecutive seizure decisions to remove the effect of transient noises. Figure. 10 shows the latency among patients, with an average of 1.1s.

F. Feature Importance

Figure. 11 summarizes the overall performance of examined features across patients. Line-length stands out as the best feature, in accordance with many other studies [37]. Variance, ripple, and fast ripple are next. Interestingly, we observe a clear shift in discriminative performance of spectral power features from Berger bands toward gamma, ripple, and fast ripples (all normalized). However, as explained in [15], [9], to distinguish between seizure and non-seizure data, both dominant and less dominant frequency components are required, as well as the spatial variation among channels, that is achieved through a multichannel analysis. In this work, we implement a programmable filter with flexible bandwidth settings to cover all seizure-related frequency components. By using a single filter architecture with programmable bandwidth, the hardware complexity of FEE is significantly reduced compared to prior works that integrate multiple parallel bandpass filters.

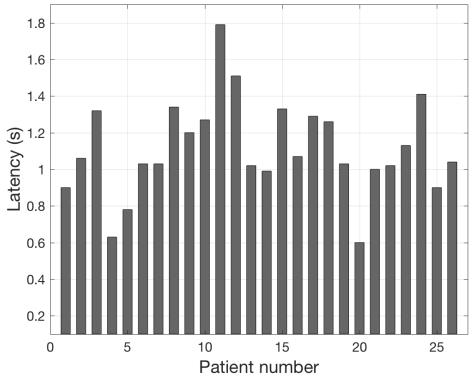


Fig. 10: The detection latency of XGB-HW across patients.

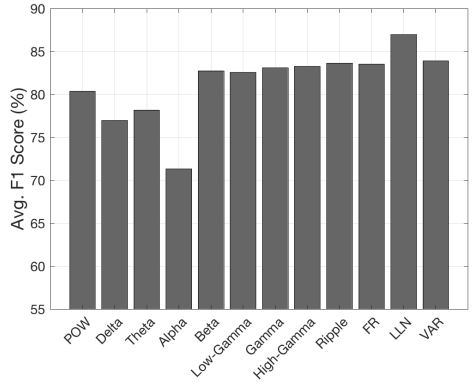


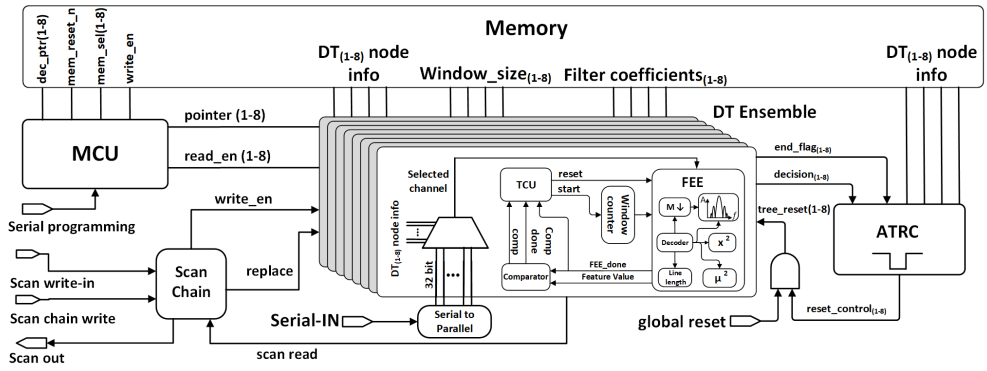
Fig. 11: Overall feature importance for the proposed classifier.

VI. SoC Implementation

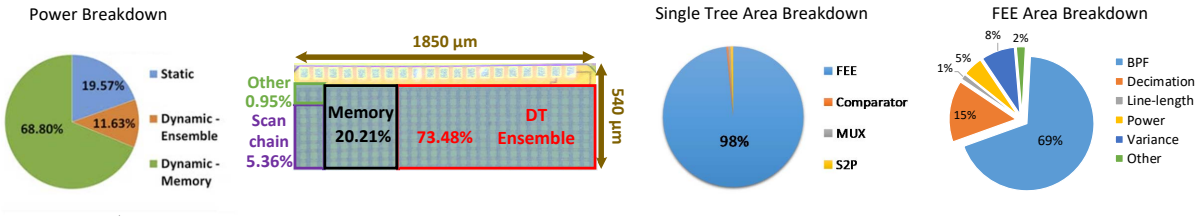
Figure. 12(a) shows the block diagram of the implemented SoC based on the asynchronous XGB classifier presented in Section IV [44], [45]. This classifier supports up to 32 neural channels. One fully programmable feature extraction unit is used per tree and controlled by the Tree Control Unit (TCU) to extract epileptic biomarkers. A Mealy FSM implementation of the closed-loop system is chosen, that substantially reduces the power and area overhead. To extract spectral density features, a single FIR filter structure is used and its coefficients are multiplexed according to the feature being processed, thus reducing the total area. As a result, the classifier achieves an energy efficiency of 41.2nJ/class in a small area of $1\mathrm{mm}^2$.

Features of line-length, variance, and total power are implemented with standard digital logic according to their mathematical definitions in Table III, and contribute to a small portion of feature extraction area (<15%), as shown in Fig. 12(b). The main blocks of the implemented Mealy FSM include the ensemble of 8 DTs with programmable FIR filters, a Memory Control Unit (MCU), and an Asynchronous Tree

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems



(a)



(b)

Fig. 12: (a) Block diagram of the implemented SoC; (b) Power breakdown, die micrograph, and area breakdown of a single tree and FEE.

Reset Control (ATRC). The detailed functional description of these blocks is discussed as follows.

1) DT Ensemble: The ensemble includes 8 decision tree structures with a maximum depth of 4 (15 nodes). For each tree, TCU sets the next state's memory pointer according to the current state, comparator status, and internal flags. A multiplexer selects one channel from the 32-channel input data, according to the current state. This channel is then fed to FEE. At the last processing node of each tree, TCU sends out the 'tree-end' flag as well as final node info to ATRC. Epileptic features are computed in the FEE module. A decoder activates/deactivates its sub-modules according to the feature under study at the current node.

2) Programmable FIR Filters: To calculate spectral power features, a cascade of two FIR stages is implemented. The first stage decimates input samples, while the second stage provides bandpass filtering. Each stage may be bypassed according to selected feature. Since at each node of a tree only one feature is being processed, a single filter structure with programmable coefficients can be used. This would significantly relax the area-power constraints in feature extraction module. The FIR filters have Type-I direct symmetric structures with 7 and 35 taps for the first and second stages, respectively. A direct symmetric structure enables using half the multipliers needed for a standard FIR filter, as well as 50% saving in coefficient memory. A high number of taps would lead to extra power and area in FEE and memory. To select optimal number of taps, extensive analysis was made. Given the importance of higher frequency features in seizure detection as shown in Figure. 11, we particularly focused on the required accuracy for capturing low-amplitude ripple and fast ripple features (i.e., HFOs) with short duration and rare occurrence [31], [26]. Thus, the filter architecture and length were chosen to ensure lower than 5% error in HFO extraction over the entire training set.

3) Memory Control Unit: MCU monitors the read/write access to the memory. In the write mode, a decoder activates different memory sub-modules for programming through the serial input, that is generated during patient-specific training. The filter coefficients and prediction model are stored in memory. The fully programmable memory allocation enables a patient-specific seizure detection. The total size of the register type memory is less than 1kB, with shared filter coefficients using 228B. The memory associated with filter coefficients is shared among trees. Thus, it is not scaled by increasing the number of trees. Each DT has a dedicated 690b of memory for its node information (690B for 8 trees). Four sub-memory blocks with a depth of 15 store the tree structure, including each node's feature/channel selection, decimation filter selection, threshold, and leaf values, tree structure (whether there is a child node or not), and window size for feature extraction.

In the read mode, MCU receives pointer address and commands from each DT, and sends back the requested information. It also activates/deactivates the associated filter coefficients from memory to DTs, according to the corresponding node info. Trees work independently in a parallel fashion, using an Asynchronous Tree Reset Control.

4) Asynchronous Tree Reset Control: To effectively capture all abnormalities in the data, each tree works independently and computes its trained features to maximize the accuracy, regardless of computational delay. When the 'tree-end' flag of a tree is raised, ATRC stores the tree status and resets it to the initial state. After reset is cleared, the tree starts processing of new input data. ATRC holds the tree status until the next available 'tree-end' flag. Finally, ATRC assigns each tree's

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

respective leaf values to calculate D_{final} according to (6).

Input Precision: The input bit precision should be sufficiently high to ensure the detectability of weak high-frequency features. According to [46], at least 12-bit resolution is required to extract correct FR patterns for seizure onset detection. On the other hand, lower bit resolution is preferred to reduce the chip area and power. To find the required number of bits, HFOs from various patients were calculated at 9-12 bit precisions of input data, and compared to those extracted from ideal floating point input. With some extra margin that accounts for lower effective resolution of ADC, we chose 12 bits that ensures less than 0.1% error in the amplitude of HFOs.

Experimental Setup and Measurement Results: The chip micrograph of the proposed classification architecture fabricated in a 65nm TSMC process and its area breakdown are depicted in Fig. 12(b), as well as the area breakdown of a single tree and the FEE. Each tree, including its dedicated and shared memory units, takes 11.25% of the die area. Figure. 12(b) also shows the power breakdown of the proposed SoC operating at a 0.8V supply, with an energy efficiency of 41.2nJ/class. Power measurements were made at worst-case scenarios where all the internal registers are switching and FEE is saturated (i.e., electrical onset of seizure is approaching).

In order to test the seizure detection performance of the fabricated chip, iEEG recordings from epileptic patients were digitized on a local PC with 12-bit resolution. The digitized data of all channels were then serialized and stored on the DDR2 SDRAM of an Altera DE4 board, as shown in Fig. 13. The information of prediction model was serially sent to the Serial Programming input of the implemented SoC (shown on the right). Once the prediction model is stored on memory, FPGA provides input clock and start command to SoC. For each patient, the chip is programmed according to the ensemble structure of his/her trained prediction model. Then, the test iEEG data of that patient is loaded to the chip for feature extraction and classification. Using the measured decisions, sensitivity and specificity are calculated. We tested the chip with 2253 hours of iEEG data from 20 patients. As the chip handles up to 32 input channels, those patients with up to 32 channels in their trained prediction model were used for the test. Given the limited data storage on FPGA, up to 10 hours of iEEG data was used for each test. The exact duration was determined based on the state of iEEG data. In the case of significant seizure-like activity in the vicinity of 10 hour, the duration of test data was reduced to 9 hours, with the last 1-hr added to the following experiment. Table IV summarizes the performance of this system compared to the state-of-the-art on-chip classifiers for seizure detection. In measurements, the classifier achieves an average sensitivity and specificity of 83.7% and 88.1%, respectively. For a fair comparison with state-of-the-art, energy and area of [16] are normalized to the 65nm technology node. The proposed architecture achieves over 27× improvement in energy-area-latency product.

VII. Scalability and Hardware Optimization

The small number of channels in existing neural interface technology remains a barrier to the therapeutic potential. For instance, the spatial coverage and resolution of electrodes has a high impact on the detection accuracy of epilepsy implants [29]. The proposed XGB classifier in this paper is inherently scalable to multi-sensor and multichannel operation, through sharing the computational and memory resources for feature extraction and classification among channels. In contrast to a majority of other classifiers that linearly scale in computational and memory requirements with number of channels and features, the proposed classifier computes a handful of features per tree, regardless of total channel count. This approach enables significant savings in computational resources and required storage on chip.

Although we have chosen a relatively simple feature set in this study, one may use additional complex and non-linear features to boost the accuracy at a negligible cost. The total number of feature extraction units to be physically placed on chip is proportional to number of trees, while only one feature is computed in each tree at a time, saving both power and area. In other words, we can include as many features as the application requires, since they only scale up with number of trees and do not pose excessive memory and hardware requirements. Without any channel selection or feature reduction techniques (that is required in most traditional methods due to large dimension of features), the proposed classifier inherently selects an optimal set of channels and related features that form the tree structure. Thus, the main contribution of this work is a hardware approach to enable energy reduction by minimizing the number of simultaneously extracted features,

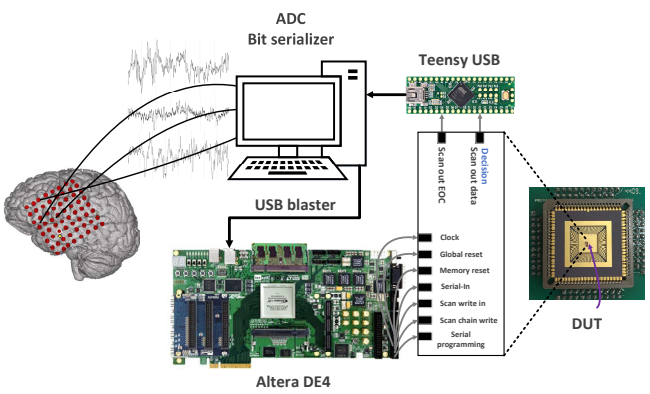


Fig. 13: Experimental setup to measure the on-chip classifier.

TABLE IV: SoC Performance and Comparison

Parameter	ISSCC'13 [16]	JSSC'13 [15]	JSSC'14 [17]	JSSC'13 [18]	This work
Process	180 nm	180 nm	180 nm	130 nm	65 nm
Classifier	Non-Lin SVM	Lin-SVM	LLS	SVM‡	XGB
Signal Modality	EEG	EEG	iEEG	EEG	iEEG
Channel Count	8	8	8	18	32
Energy Eff.	1.23*μJ/class	1.52*μJ/class	77.91μJ/class	273μJ/class	41.2nJ/class
Logic Size†	2.27M	3.3M	N. A.	371k	330k
Memory [kB]	N.A.	N.A.	N.A.	32§	1
Area	7 mm^2*	8.18 mm^2*	6.5 mm^2*	5.13 mm^2	1 mm^2
Sensitivity [%]	95.1	N.A.	92	N.A.	83.7
Specificity [%]	94	N.A.	N.A.	N.A.	88.1
Latency [s]	2	2	0.8	N.A.	1.79††

* Area and Energy Efficiency conservatively estimated from A/P breakdown
† Number of equivalent NAND2 gates with driving strength of one
‡ Linear, Polynomial, RBF
§ 32kB SV MEM, 16kB Programming MEM, 16kB Data MEM
†† Worst case latency (patient 11)

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

thus breaking the energy-area vs. accuracy tradeoff. Bufferless processing of data in a closed-loop scheme is employed, and programmable bandpass filters further decrease the overall area overhead. The total power can be further reduced by dynamically controlling the channel activation and powering down the low-noise amplifiers in unused channels.

A. Energy-Quality Tradeoffs and Scaling

In our proposed gradient-boosting classifier, each tree contributes to roughly 10% of total power (static and dynamic). Based on the performance curves shown in Fig. 8, we chose to implement an ensemble of eight trees with a maximum depth of four, to achieve an average AUC of more than 90% across a large population of patients with varying number of electrodes, seizures, and sampling rates. However, not all patients in our database need as many trees for an accurate discrimination of their seizures, as depicted in Fig. 14 (top curves). Therefore, we enabled a programmable on/off control for each tree in the ensemble, so that upon a patient-specific training phase, one or more trees could be switched off to save power, with a minimum impact on quality. In other words, depending on the difficulty of detection task, the required number of trees can be switched on to achieve an expected classification accuracy (e.g., eight trees for patients with hardly detectable seizures, such as patient 24 in Fig. 14). We use the AUC as our quality metric, that is widely used to evaluate the predictive accuracy of a classifier.

Boosting methods generally attain high discrimination by sequential training of weak classifiers. Here, the XGB attempts to increase the predictive accuracy by making a more accurate prediction at each iteration [36]. However, increasing the number of DTs increases the memory and power requirements of the system. The proposed XGB hardware is inherently quality-scalable through programming the number and depth of the active trees, with a maximum depth set at four. Moreover, our design offers a unique flexibility to accommodate various tree structures specific to each patient, to trade the predictive accuracy with energy (i.e., avoid unnecessary energy dissipation when accuracy is just enough for a patient). We explored the hardware parameters of tree count and depth across all patients, as potential knobs for energy-quality scaling.

As shown in Fig. 14, we observe that in most patients, a small number of trees are sufficient for a reliable seizure detection. Indeed, the structure of successive trees are very similar in most patients, and by switching off the last few trees, we only observe a slight decrease in predictive accuracy. While chip area is limited by the required number of trees for worst case patients, the energy usage can be scaled for cases with easily detectable seizures. The other alternatives (knobs) for energy-quality scaling include pruning of trees, or forcing the algorithm to use energy-aware features by modifying the cost function (i.e., adding an energy constraint similar to the delay constraint in Fig. 6). However, we specifically observed that for most patients, the very last 3–4 trees in the iterative training process of XGB have a slight impact on performance and could even cause overfitting. In addition, our proposed asynchronous approach requires a single FEE in each tree that freely runs to compute one feature at a time. Thus, its energy is less sensitive to the depth parameter and is rather controlled by sampling frequency. Thus, we have focused on the hardware knob of tree count, that is easily integrated into our power-aware classification prototype.

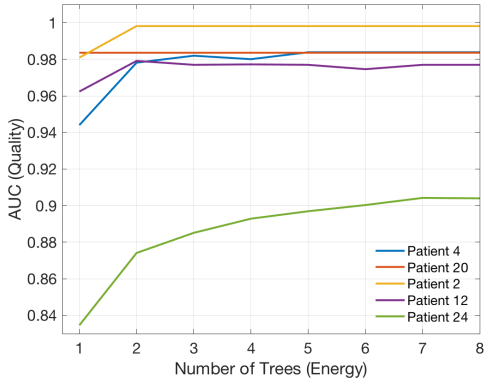


Fig. 14: Measured AUC versus number of trees for various patients.

B. Discussion on Hardware Optimization

Various opportunities to improve the energy and area efficiency of proposed classifier could be explored that remain as a future work. For instance, the input bit precision in our chip implementation has been chosen sufficiently high to allow the detectability of high-frequency features. Given the inherent error tolerance in machine learning algorithms, the energy per classification can be reduced by relaxing the quality or precision of features. For low-power and compact implementation in particular, reducing the resolution of coefficients in filter banks, feature thresholds, and leaf values is critical. New approaches to train decision trees with fixed-point and low-cost parameters can be investigated, similar to the works that reduce precision in DNNs [3], SVMs and LRs [2]. Since the training is usually performed offline, the associated cost is not critical. Such parameters could further be used as potential knobs in the proposed energy-quality scaling framework.

Furthermore, DTs can be trained to incorporate the costs of misclassification (FP or FN) and feature computation (power, area, delay) in the tree induction process. For example, it is critical to achieve a high sensitivity in seizure detection, while keeping the false alarm rate and latency below a tolerable level. This can lead to development of cost-sensitive decision trees, where the top-down tree induction algorithm may be adapted to maintain a pre-specified cost, therefore trading off the unnecessary accuracy (e.g., very high specificity or low latency) and energy. Besides, using various design parameters of DTs, the XGB classifier can be programmed to trade energy and quality in a structured and dynamic fashion.

VIII. Conclusions

In this work, we addressed the challenge of designing a low-power machine learning algorithm for on-chip neural data classification. We proposed a novel hardware architecture for a gradient-boosted decision tree model, with a single feature extraction engine and programmable FIR filter per tree. The proposed asynchronous tree operation enables efficient classification of multichannel neural data, with significantly lower

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

memory, power and area requirements compared to state-of-the-art. As a result, this on-chip classifier achieves an energy-area-latency product that is 27× lower than prior works, while processing the highest number of channels. The hardware architecture, design optimization and tradeoffs are discussed, and algorithm performance based on proposed model and SoC measurements is presented. Such classifiers could potentially allow full integration of processing circuitry with the sensor array in various resource-constrained biomedical applications.

REFERENCES

[1] C. Bishop, "Pattern Recognition and Machine Learning (1st. ed.)," *Springer*, New York, NY., 2006.

[2] H. Albalawi, Y. LI, and X. LI, "Training Fixed-Point Classifiers for On-Chip Low-Power Implementation," *ACM Transactions on Design Automation of Electronic Systems*, vol. 22, no. 69, 2017.

[3] V. Sze, "Designing Hardware for Machine Learning: The Important Role Played by Circuit Designers," *IEEE Solid-State Circuits Magazine*, vol. 9, no. 4, pp. 46-54, 2017.

[4] B. Murmann, D. Bankman, E. Chai, D. Miyashita, and L. Yang, "Mixed-signal circuits for embedded machine-learning applications," *Proc. Asilomar Conf.*, 2015, pp. 1341-1345.

[5] Available online at www.kaggle.com/c/seizure-detection

[6] M. Alioto, "Energy-quality scalable adaptive VLSI circuits and systems beyond approximate computing," *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 127-132, 2017.

[7] T. L. Skarpaas and M. J. Morrell, "Intracranial stimulation therapy for epilepsy," *Neurotherapeutics*, vol. 6, pp. 238-43, 2009.

[8] W. C. Stacey and B. Litt, "Technology insight: neuroengineering and epilepsy–designing devices for seizure control," *Nature Clinical Practice Neurology*, pp. 190-201, 2008.

[9] A. Shoeb, H. Edwards, J. Connolly, B. Bourgeois, S. T. Treves, J. Guttag, "Patient-specific Seizure Onset Detection," *Epilepsy & Behavior.* vol. 5, pp. 483-498, 2004.

[10] N. Verma, A. Shoeb, J. Bohorquez, J. Dawson, J. Guttag, A. P. Chandrakasan, "A Micro-Power EEG Acquisition SoC With Integrated Feature Extraction Processor for a Chronic Seizure Detection System," *IEEE J. Solid-State Circuits*, vol. 45, pp. 804–16, 2010.

[11] M. Shoaran, C. Pollo, K. Schindler, A. Schmid, "A Fully-Integrated IC with 0.85-μW/Channel Consumption for Epileptic iEEG Detection," *IEEE Trans. Circuits Sys. II: Express Briefs*, vol. 62, pp. 114-118, 2015.

[12] M. Shoaran, M. Shahshahani, M. Farivar, J. Almajano, A. Shahshahani, A. Schmid, A. Bragin, Y. Leblebici, A. Emami, "A 16-Channel 1.1mm^2 Implantable Seizure Control SoC with Sub-μW/Channel Consumption and Closed-Loop Stimulation in 0.18μm CMOS," *IEEE Symp. VLSI Circuits (VLSIC)*, HI, 2016.

[13] M. A. B. Altaf, C. Zhang, J. Yoo, "A 16-ch patient-specific seizure onset and termination detection SoC with machine-learning and voltage-mode transcranial stimulation," *Int. Solid-State Circuits Conf. (ISSCC)*, 2015.

[14] C. Zhang, M. A. B. Altaf, J. Yoo, "Design and Implementation of an On-Chip Patient-Specific Closed-Loop Seizure Onset and Termination Detection System," *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 4, pp. 996-1007, 2016.

[15] J. Yoo, L. Yan, D. El-Damak, M. A. B. Altaf, A. Shoeb, A. P. Chandrakasan, "An 8-Channel Scalable EEG Acquisition SoC With Patient-Specific Seizure Classification and Recording Processor," *IEEE J. Solid-State Circuits*, vol. 48, pp. 214–28, 2013.

[16] M. A. B. Altaf, J. Tillak, Y. Kifle, J. Yoo, "A 1.83 μJ/classification non-linear support-vector machine-based patient-specific seizure classification SoC," *IEEE Int. Solid-State Circuits Conf. (ISSCC)*, pp. 100-101, 2013.

[17] W. M. Chen et al., "A fully integrated 8-channel closed-loop neural-prosthetic SoC for real-time epileptic seizure control," *IEEE J. Solid-State Circuits*, vol. 49, no. 1, pp. 232–247, 2014.

[18] K. H. Lee and N. Verma, "A Low-Power Processor With Configurable Embedded Machine-Learning Accelerators for High-Order and Adaptive Analysis of Medical-Sensor Signals," *IEEE J. Solid-State Circuits*, pp. 1625-1637, 2013.

[19] Z. Wang, R. Schapire, N. Verma, "Error-adaptive classifier boosting (EACB): Leveraging data-driven training towards hardware resilience for signal inference," *IEEE Trans. Circuits Syst. I (TCAS-I)*, vol. 62, no. 4, pp. 1136-1145, 2015.

[20] T. C. Chen et al., "1.4 μW/channel 16-channel EEG/ECoG processor for smart brain sensor SoC," *IEEE Symp. VLSI Circuits*, pp. 21-22, 2010.

[21] A. Page, C. Sagedy, E. Smith, N. Attaran, T. Oates and T. Mohsenin, "A Flexible Multichannel EEG Feature Extractor and Classifier for Seizure Detection," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 62, no. 2, pp. 109-113, 2015.

[22] T. Roh, K. Song, H. Cho, D. Shin, H. J. Yoo, "A Wearable Neuro-Feedback System With EEG-Based Mental Status Monitoring and Transcranial Electrical Stimulation," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 8, no. 6, pp. 755-764, 2014.

[23] S. Y. Hsu, Y. Ho, P. Y. Chang, C. Su and C. Y. Lee, "A 48.6-to-105.2μW Machine Learning Assisted Cardiac Sensor SoC for Mobile Healthcare Applications,," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 4, pp. 801-811, April 2014.

[24] Available online at www.ieeg.org

[25] J. Zhang, L. Huang, Z. Wang, N. Verma, "A seizure-detection IC employing machine learning to overcome data-conversion and analog-processing non-idealities," *IEEE Custom Integrated Circuits Conf.*, 2015.

[26] A. Bragin, J. Engel, C. L. Wilson, I. Fried, G. Buzsaki, "High-frequency oscillations in human brain," *Hippocampus*, vol. 9, pp. 137-142, 1999.

[27] M. Ayinala and K. K. Parhi, "Low complexity algorithm for seizure prediction using Adaboost," *Int. Conf. of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 1061-1064, 2012.

[28] M. Bandarabadi, A. Dourado, C. Teixeira, T. Netoff, K. Parhi, "Seizure Prediction with Bipolar Spectral Power Features using Adaboost and SVM Classifiers," *Int. Conf. of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2013.

[29] M. Stead, M. Bower, B. H. Brinkmann, K. Lee, W. R. Marsh, F. B. Meyer, B. Litt, J. V. Gompel and G. A. Worrell, "Microseizures and the spatiotemporal scales of human partial epilepsy," *Brain*, 2010.

[30] L. Breiman, J. Friedman, R. Olshen, C. Stone, "Classification and Regression Trees," *Wadsworth*, 1984.

[31] M. Shoaran, M. Farivar, A. Emami, "Hardware-Friendly Seizure Detection with a Boosted Ensemble of Shallow Decision Trees," *Int. Conf. of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2016.

[32] A. Benbasat and J. Paradiso, "A Framework for the Automated Generation of Power-Efficient Classifiers for Embedded Sensor Nodes," *ACM SenSys*, 2007.

[33] Y. Yang, S. Boling and A. J. Mason, "A Hardware-Efficient Scalable Spike Sorting Neural Signal Processor Module for Implantable High-Channel-Count Brain Machine Interfaces," *IEEE Transactions on Biomedical Circuits and Systems*, vol. 11, no. 4, pp. 743-754, Aug. 2017.

[34] K. M. H. Badami, S. Lauwereins, W. Meert, M. Verhelst, "A 90 nm CMOS 6 μW power-proportional acoustic sensing frontend for voice activity detection," *IEEE J. Solid-State Circuits*, pp. 291-302, 2016.

[35] J. H. Friedman, "Greedy function approximation: a gradient boosting machine", *Annals of Statistics*, pp.1189-1232, 2001.

[36] T. Chen, T. He, "xgboost: eXtreme Gradient Boosting," *R package version 0.4-2*, 2015.

[37] L. Logesparan, A. J. Casson, E. Rodriguez-Ville, "Optimal features for online seizure detection," *Med Biol Eng Comput.*, pp. 659-669, 2012.

[38] R. Esteller, J. Echauz, T. Tcheng, B. Litt, B. Ples, "Line length: an efficient feature for seizure onset detection," *Int. Conf. of IEEE Eng. in Medicine and Biology Society (EMBC)*, pp. 1707-1710, 2001.

[39] K. Schindler, H. Leung, C. E. Elger, K. Lehnertz, "Assessing seizure dynamics by analysing the correlation structure of multichannel intracranial EEG," *Brain*, vol. 130, pp. 65–77, 2007.

[40] P. E. McSharry, L. A. Smith, L. Tarassenko, "Comparison of Predictability of Epileptic Seizures by a Linear and a Nonlinear Method," *IEEE Transactions on Biomedical Engineering*, vol. 50, pp. 628–633, 2003.

[41] K. K. Majumdar and P. Vardhan, "Automatic Seizure Detection in ECoG by Differential Operator and Windowed Variance," IEEE Trans. Neural Syst. Rehabil. Eng., vol. 19, pp. 356–65, 2011.

[42] L. Ayoubian, H. Lacoma, J. Gotman, "Automatic seizure detection in SEEG using high frequency activities in wavelet domain," *Med Eng Phys.*, vol. 35, no. 3, pp. 319–328, 2013.

[43] M. Bandarabadi, C. A. Teixeira, J. Rasekhi, and A. Dourado, "Epileptic seizure prediction using relative spectral power features," *Clin Neurophysiol.*, 2014.

[44] M. Taghavi, B. A. Haghi, M. Farivar, M. Shoaran, A. Emami, "A 41.2 nJ/class, 32-channel on-chip classifier for epileptic seizure detection," *Int. Conf. IEEE Eng. Medicine and Biology Society (EMBC)*, 2018.

[45] M. Shoaran, B. A. Haghi, M. Farivar, A. Emami, "Efficient Feature Extraction and Classification Methods in Neural Interfaces," *the Bridge, National Academy of Engineering*, vol. 47, no. 4, pp. 31-35, Winter 2017.

[46] C. Qian, J. Shi, J. Parramon, E. Sanchez-Sinencio, "A Low-Power Configurable Neural Recording System for Epileptic Seizure Detection" *IEEE Transactions on Biomedical Circuits and Systems*, 2013.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JETCAS.2018.2844733, IEEE Journal on Emerging and Selected Topics in Circuits and Systems

Mahsa Shoaran is an Assistant Professor in the School of Electrical and Computer Engineering at Cornell University and director of Cornell Neuroengineering Laboratory. Prior to joining Cornell, she was a Postdoctoral Fellow in Electrical Engineering and Medical Engineering at the California Institute of Technology. She received her PhD from Swiss Federal Institute of Technology (EPFL) in 2015, and her B.Sc. and M.Sc. from Sharif University of Technology in 2008 and 2010, respectively. Mahsa is a recipient of both Early and Advanced Swiss National Science Foundation (SNSF) Postdoctoral Fellowships. She was named a Rising Star in EE/CS by MIT in 2015. Her main research interests include low-power circuit and system design for biomedical applications, brain-computer interfaces, biomedical signal processing, embedded classification and machine learning, and neuromodulation therapies for neurological disorders.

Benyamin Allahgholizadeh Haghi received the B.S. degree in electrical engineering and mathematics from Sharif University of Technology, Tehran, Iran, in 2016 and the M.S. degree in electrical engineering from the California Institute of Technology (Caltech) in 2018, and in currently working toward the Ph.D. degree at the California Institute of Technology (Caltech). In summers 2014 and 2015, he was an undergraduate visiting student researcher in Prof. Amin Arbabian's research group at Stanford University, where he focused on developing an interferogram-based breast tumor classification algorithm and implementing a fast iterative reconstruction algorithm for microwave-induced thermoacoustic imaging, respectively. He started working on seizure detection using machine learning based approach when he arrived at Caltech. His research focuses on the design and implementation of robust and high performance closed-loop brain machine interface algorithms and the circuits/technologies that compromise them. Mr. Haghi was the recipient of the 2017 Caltech Departmental Fellowship, and the 2018 Chen Institute for Neuroscience Fellowship.

Milad Taghavi was born in Tehran, Iran. He received the B.S. degree in electrical engineering from Sharif University of Technology, Tehran, Iran, in 2013. He received the M.S. degree from University of Washington, Seattle, in 2014 with highest honors. He won the "Best Presentation Recognition Award" in IECON 2014, Texas, Dallas, for the design of bidirectional single phase chargers for PHEVs. In 2016, he received his M.S. degree in electrical engineering from California Institute of Technology. Currently, he is working toward his PhD degree in Electrical Engineering at Caltech, with special emphasis in analog and mixed-signal circuits and systems.

Masoud Farivar received his PhD from California Institute of Technology in 2016, and his dual degree B.Sc. from Sharif University of Technology in Electrical Engineering and Computer Science. He is currently a research scientist at Google, Mountain View. Previously, he was a postdoctoral fellow in Medical Engineering at the California Institute of Technology, working on machine learning solutions for implantable devices. His area of interest includes power systems optimization/control, and machine learning. He was a recipient of Amgen Postdoctoral fellowship and the Resnick Institute fellowship for sustainability.

Azita Emami is the Andrew and Peggy Cherng Professor of Electrical Engineering and Medical Engineering at Caltech, and a Heritage Medical Research Institute Investigator. She also serves as the Executive Officer (Department Head) for Electrical Engineering. She received her M.S. and Ph.D. degrees in Electrical Engineering from Stanford University in 1999 and 2004 respectively, and her B.S. degree from Sharif University of Technology in 1996. From 2004 to 2006 she was with IBM T. J. Watson Research Center before joining Caltech in 2007. Her current research interests include integrated circuits and systems, integrated photonics, wearable and implantable devices for neural recording, neural stimulation, sensing and drug delivery. She is currently the associated editor for the IEEE Journal of Solid State Circuits (JSSC). She is also an IEEE SSCS distinguished lecturer.