Sketching-based High-Performance Biomedical Big Data Processing Accelerator

Amey Kulkarni, Ali Jafari, Chris Sagedy, and Tinoosh Mohsenin
Department of Computer Science & Electrical Engineering
University of Maryland, Baltimore County

***Abstract*—Multi-Sensor health monitoring systems are used to predict near future events of our health system. Each sensor generates humongous amount of data per second and needs to be processed in real-time. At the same time health monitoring systems are battery operated, thus they have rigid constraints on power and area of processing platform. Additionally, health monitoring systems should be accurate, thus we adapt machine learning techniques to improve detection accuracy. We propose a programmable Big Data Processing framework to reduce on-chip communications and computations, thus reducing energy of the processing. We integrate a low-overhead sketching framework with a low-power programmable PENC many-core platform. The sketching technique reduces the data communications and computations, additionally processing time is scaled down by parallel processing on the many-core platform. For demonstration we show seizure detection application with 22-channel of electroencephalograph (EEG), each channel generates 256 samples per second requiring total of 88 Kbps data rate. The computations are reduced by 16× while energy consumption of processing is reduced up to 68%. For compression rates of 2-16×, the seizure detection performance for sensitivity and specificity is degraded by 2.07% and 2.97%, respectively for Logistic Regression classifier.**

Big Data Processing, Sketching Technique, Many-Core, Seizure Detection

I. Introduction

Real-time applications such as health monitoring and video surveillance generate humongous amount of data every hour at unprecedented rate. The generated big data sets are used for minimizing risk, identifying unknown objects and uncover hidden patterns. To exploit hidden data patterns and achieve efficient predictions, Machine Learning (ML) techniques are adopted. For example, in smart and wearable health monitoring devices, ML techniques are adopted to predict future health hazards such as increased blood pressure, blood-glucose levels, fall detection and seizure detection [1],[2],[3]. To provide efficient health monitoring a multi-sensor approach is adopted, where each sensor produces large number of samples per second. At every sample period, monitoring device needs to process several DSP and ML kernels. Thus stringent real-time constraints demand fast processing platform whereas wearable battery operated device necessitates an extremely energy efficient platform. Personalized biomedical devices typically consists of three main circuit blocks: 1. Signal Acquisition block including analog to digital converter, 2. Digital Signal Processing block which typically contains Feature extractor and ML Classifiers 3. Radio transmitter to transmit the processed data or prediction to the user or medical personnel. In this paper,

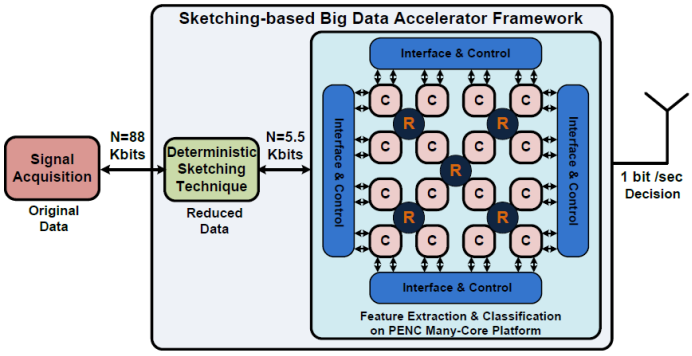

Fig. 1. Proposed Sketching-based Big Data Acceleration Framework, where C- Cluster of cores, R- Hierarchical router

we propose a sketching-based big data processing framework that performs all signal processing locally at the sensor. We integrate a low-overhead sketching framework with domain specific programmable PENC many-core accelerator to curtail computation power [4]. Whereas Sketching technique at the front end reduces number of data transfers, computations, and storage. The sketching algorithm reduces data by obtaining linear combination of the data before processing. Data reduction by adopting Sketching-based framework causes two important challenges that need to be addressed 1. optimal classification error 2. low hardware overhead for Sketch implementation. To demonstrate efficiency of the framework, we employ the proposed framework for multi-channel seizure detection application. A multi-channel seizure detection monitoring system generates enormous amount of EEG data, which needs to be processed on-chip in real-time with low power consumption.

Feature extraction and classification is performed locally at the sensor [1]. Thus, it transmits only one bit prediction i.e seizure or no-seizure, this reduces transmission power significantly. In addition, transmitter can be in sleep mode while processing. Figure 1 shows proposed Sketching-based big data processing framework. First, Deterministic Sketching technique is adopted for data reduction. Then the sparse data set is fed to the feature extraction and classification module implemented on programmable PENC many-core platform. The sketching-based framework reduces number of input samples for feature extraction thus reducing processing time. Finally, the prediction is transmitted to the user or medical personnel. Assuming AFE takes 256 samples 16 bits each, the transmitter rate can potentially change from 88 Kbits/s (for 22-channels)

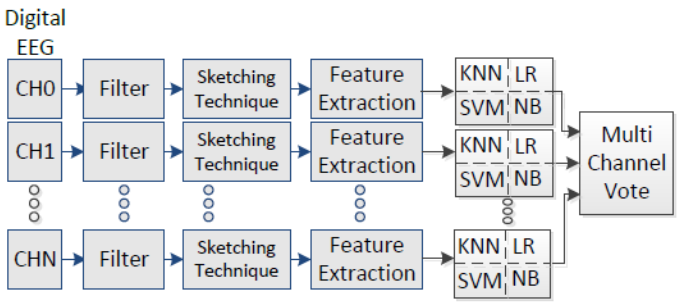

Fig. 2. Block diagram of the multi-channel seizure detection processor containing Sketching kernel (to reduce data rate), feature extraction, ML classifier, and multi-channel vote

to 1 bit/s.

The main contributions to the proposed work include:

- Sketching technique to reduce data and analyze classification accuracy with different sketching rates.
- Hardware overhead analysis of Sketching technique for Seizure detection on PENC many-core platform.
- Parallel implementation of ML algorithms on PENC many-core accelerator and performance evaluations in terms of energy consumption and processing time.

II. Sketching-based Framework for Big Data Processing

In this section, we briefly discuss sketching algorithm, feature extraction, ML algorithms classification and seizure detection approach.

A. Sketching Algorithm

In Sketching, the data is acquired at a rate proportional to information rate, i.e it obtains linear combinations of the data rather than reducing the number of samples [5]. The basic theory behind Sketching lies in solving equation 1. Let ϕ be the measurement matrix of dimension $M \times N$, where M is the number of measurements to be taken and N is the length of the signal and x be a m-sparse signal of length N. Multiplying these two vectors yields y of length M, which contains the measurements obtained by the projection of x into ϕ.

$$Y = \phi X \tag{1}$$

Traditionally Measurement matrix ϕ is generated using Bernoulli or *iid* Gaussian process which satisfy Restricted Isometric Property (RIP) with high probability [6]. However, constructing Bernoulli or *iid* Gaussian matrices in hardware is complex. Therefore we adopt deterministic random matrix to decrease hardware overhead in terms of area and energy requirements. Deterministic random matrix (DRM) is built by randomly choosing a subset of the rows of an identity matrix [7]. Figure 4D shows a reconfigurable sketching architecture using a DRM technique for different input signal sizes. A block RAM is used to store one window of input signal for sampling purpose. Additionally, a small ROM is used to store the locations of non-zero entries of deterministic random matrix. Indeed, in this method there is no need to save all zeros and ones in the ROM. In previous work [1], the authors proved

TABLE I

Formulas for the reduced 5 features: Area under the wave, normalized decay, line length, avg peak amplitude, and avg valley amplitude.

W = window length, x = input, P = # peaks, V = # valleys

Area Under Curve	Normalized Decay	
$A = \frac{1}{W}\sum_{i=0}^{W-1} x_i$	$D = \lvert \frac{1}{W-1}\sum_{i=0}^{W-2} I(x_{i+1} - x_i < 0) - .5 \rvert$	
Line Length	Avg Peak Amplitude	Avg Valley Amplitude
$\ell = \sum_{i=1}^{W} -1\lvert x_i - x_{i-1}\rvert.$	$P_A = \log_{10}\left(\frac{1}{P}\sum_{i=0}^{P-1} x_{p(i)}^2\right)$	$V_A = \log_{10}\left(\frac{1}{V}\sum_{i=0}^{V-1} x_{v(i)}^2\right)$

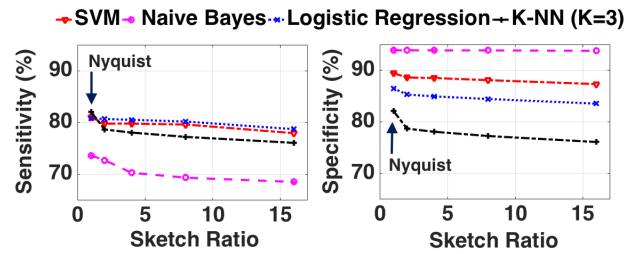

Fig. 3. Performance of the proposed seizure detection algorithm with direct use of sketched signals and employing a deterministic random matrix. Degradation in sensitivity and specificity is approximately 2.07% and 2.97%, respectively, up to a sketch rate of 16 .

that the sketching architecture using DRM technique decreases the hardware overhead compared to sketching module using LFSR circuit [8],[9].

B. Seizure Detection Approach and Performance Analysis

The data used to evaluate the performance of the proposed system consists of EEG recordings are collected at the Children's Hospital Boston [10]. Figure 2 shows the block diagram for the Seizure Detection with integration of sketching-based framework. The data obtained from EEG scalp is in the form of raw time series, applying raw data to classifiers will result in low accuracy. Thus the EEG sensor data is passed through a filter to remove high frequency and DC components. This data is used as the input to the proposed Sketching algorithm for different Sketch Ratio (SR), up to 16x. A window of 256 samples (one second per channel) is chosen for the input of Sketching Algorithm. Then, the sketched data are used to create 5 simple features for each second of EEG data per channel [2]. In this study we consider area under curve, normalized decay, line length, average peak amplitude, and average valley amplitude as features [2], [11]. The formulas for these five features are given in Table I. The proposed DRM sketching technique reduces feature extraction computations which consequently decreases overall seizure detection time and energy consumption. Each channel's features are then classified using one of four classifiers: K-Nearest Neighbors (KNN), Support Vector Machines (SVM), Naive Bayes (NB), or Logistic Regression (LR). A final stage is then used to produce a final decision based on a multichannel voting scheme.

Performance of the seizure detector is characterized in terms of sensitivity and specificity. Sensitivity refers to the percentage of seizure onsets identified and Specificity refers to

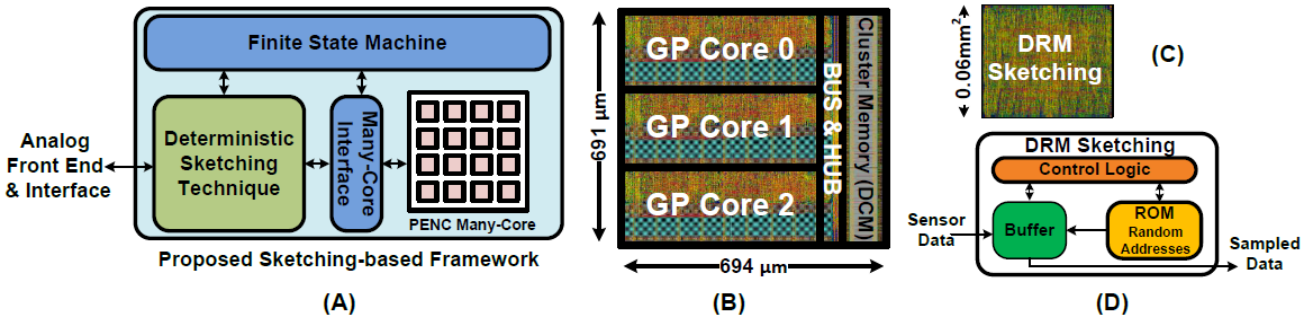

Fig. 4. (A) Block Diagram of The Proposed Big Data Accelerator for Seizure Detection Application, where Many-Core Platform performs Feature Extraction, Classification and Multi-Channel Vote. Note: SPI and AFE blocks are not implemented (B) Layout of Bus-based Cluster of PENC many-core platform in 65 nm CMOS technology, where GP is General Purpose core (C) Layout of Sketching Module in 65 nm CMOS technology (D) Block Diagram of Deterministic Random Matrix based Sketching technique

the percentage of incorrectly detected seizure onsets [1]. Also, to estimate the detector performance on the data from a patient, a leave-one record-out cross-validation approach is used [12]. The proposed system could achieve a sensitivity of 81.8% and specificity of 93.9% for Nyquist-domain seizure data (without Sketching). Figure 3 shows the seizure detector performance using sketched data for SR from 2× up to 16×. As it can be seen from the Figure 3, the degradation in sensitivity and specificity is approximately 2.07% and 2.97%, respectively, up to a sketching rate of 16×.

III. Big Data "Acceleration" Many-Core Platform

A. PENC: Many-Core Architecture

The proposed PENC many-core architecture consists of in-order processors with a 6 stage pipeline, a RISC-like DSP instruction set and a Harvard memory model. The core operates on a 16-bit data path with a minimal instruction and data memory suitable for task level parallelism. Furthermore, the core has a low complexity, reduced instruction set to further reduce area and power footprint. These processing cores can execute arithmetic, branch, and inter-core communication instructions. In this design, every three cores with a memory is grouped into a cluster which can perform intra-communication directly through a bus and inter-communication through a routing architecture. The many-core is fully implemented in verilog and placed and routed in 65nm, 1V CMOS Technology [4]. Figure 4B shows layout of bus-based cluster of PENC many-core platform.

Our many-core development environment includes an architecture simulator written in Java. The simulator serves as a reference implementation of the architecture; its purpose is to make testing, refining, and enhancing the architecture easier. It models the functionality of the processor and calculates the final state of register and data memories. It reports statistics such as the number of cycles required for ALU, branch, and communication instructions. The execution time analysis is calculated by using many-core simulator, whereas for power analysis, the algorithms are implemented on the hardware model of the many-core platform and simulated using Cadence NC-Verilog. The activity factor is then derived and is used by the Cadence Encounter tool for accurate power computation. The PENC many-core architecture is ideally suited for most biomedical applications as it addresses the characteristics and challenges inherent with these applications [4],[13][14].

B. ML Algorithm Implementations on Many-Core Platform

The seizure detection application includes both feature extraction and classification. A single core performs feature extraction on windows of samples from 22 channels in serial. It generates a test vector of five features for each window and broadcasts the results to the cores that perform classification. We explored four different supervised ML algorithms with respect to execution time, energy consumption and its storage memory requirements. For the SVM classifier, support vectors are distributed evenly among 47 cores in 16 clusters. Each core computes the dot product of the test vector with 106 support vectors and accumulates the sum of the results. After each core has completed its portion of the work, the cores communicate to compute the total sum and one core determines the final classification. The PENC must run at 160 kHz to process 22 windows of 256 samples in one second, and at 157 kHz in case of 32 samples per window. KNN is mapped on 47 cores and the PENC must run at 228 kHz without sketching and at 224 kHz in case SR = 16×. NB and LR classifiers are trained offline, thus only weight vectors are stored on cache memory. The seizure detection is then performed by taking dot product of new seizure data and weight vector.

IV. Implementation Results

The results shows that Sketching-based framework reduces computational complexity of Feature extraction kernel reducing computational latency up to 82% for SR=16 as shown in Figure 5. It also shows execution time analysis for each ML classifier with feature extraction, thus overall seizure detection time is reduced up to 68%. Table II shows energy analysis of ML algorithm mapping with feature extraction on PENC many-core platform. KNN and SVM algorithms require large number of cores and they take 10,181 and 7,056 cycles respectively to execute, thus it hides the advantage of computational complexity reduction for feature extraction kernel. However NB and LR show 24% to 68% improvement

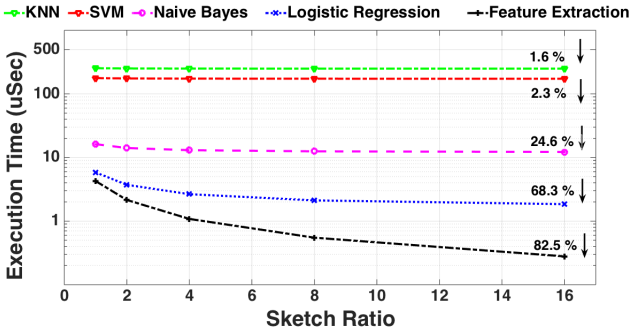

Fig. 5. Execution time analysis for the proposed system using different ML classifiers on the many-core platform

TABLE II
ENERGY ANALYSIS OF PENC MANY-CORE ML ALGORITHM MAPPING WITH THE PROPOSED BIG DATA ACCELERATION FRAMEWORK

Architecture	FE (mJ)	FE + ML Classification (mJ)			
		KNN	SVM	NB	LR
W/o Sketching	0.032	1.720	1.202	0.015	0.0057
With SR=2	0.018	1.706	1.188	0.014	0.004
With SR=4	0.011	1.699	1.181	0.013	0.0027
With SR=8	0.007	1.695	1.177	0.0122	0.0021
With SR=16	0.005	1.693	1.175	0.0119	0.0018

TABLE III
ENERGY CONSUMPTION COMPARISON OF FPGA WITH PENC MANY-CORE PLATFORM FOR THE SEIZURE DETECTION USING SKETCHING TECHNIQUE, FEATURE EXTRACTION AND ML ALGORITHMS

Architecture	SR	KNN(mJ)	SVM(mJ)	NB(mJ)	LR(mJ)
PENC Many-Core	1	1.720	1.202	0.015	0.0057
Artix-7 FPGA		1349	164.64	0.76	0.64
PENC Many-Core	16	1.693	1.175	0.0119	0.0018
Artix-7 FPGA		1348	164.29	0.40	0.29

in energy and execution time. To show the efficiency of PENC many-core platform as a Big Data accelerator, we also implemented the proposed Sketching-based framework on low power Xilinx Artix-7 FPGA platform. Great care was taken to ensure consistency of the algorithms across both platforms. For the Artix-7 FPGA implementation, the power and timing results were obtained using Xilinx XPower and Timing analyzer, respectively. Table III shows energy comparison of many-core and FPGA platforms with and without Sketching technique. For LR and NB algorithms PENC implementation consumes approximately 100× less energy as compared to FPGA implementation.

V. CONCLUSIONS

The paper proposes Big Data Processing Accelerator for Biomedical applications. The paper explored the advantages of Sketching algorithm to reduce the computational complexity, thus reducing execution time and energy. Additionally domain specific many-core platform is explored for parallel processing of commonly used ML algorithms. The sketching-based framework reduces energy of the seizure detection processor up to 68% with negligible effect on specificity and sensitivity. The proposed platform with low power cores and GALS architecture allows the platform not only to exploit the task level and data-level parallelism but also perform dynamic voltage and frequency scaling to dramatically reduce energy consumption. The ML algorithms are also implemented on low power Xilinx Artix-7 FPGA. Compared to FPGA implementation the PENC many-core, on average, consumes approximately 100× less energy.

VI. ACKNOWLEDGEMENT

Authors would like to thank Nasrin Attaran and Adam Page for some preliminary results in this work. This research is based upon work supported by the National Science Foundation under Grant No. 00008999 and 00010145.

REFERENCES

[1] A. Jafari *et al.*, "A low power seizure detection processor based on direct use of compressively-sensed data and employing a deterministic random matrix," *IEEE Biomedical Circuits and Systems (Biocas) Conference*, 2015.

[2] A. Page *et al.*, "A flexible multichannel eeg feature extractor and classifier for seizure detection," *Circuits and Systems II: Express Briefs, IEEE Transactions on*, vol. 62, no. 2, pp. 109–113, 2015.

[3] A. Page, A. Kulkarni, and T. Mohsenin, "Utilizing deep neural nets for an embedded ecg-based biometric authentication system," in *IEEE Biomedical Circuits and Systems (Biocas) Conference*, Oct 2015.

[4] A. Kulkarni, T. Abtahi, E. Smith, and T. Mohsenin, "Low energy sketching engines on many-core platform for big data acceleration," in *Proceedings of the 26th Edition of the Great Lakes Symposium on VLSI*, ser. GLSVLSI '16. New York, NY, USA: ACM, 2016.

[5] D. L. Donoho, "Compressed sensing," *Information Theory, IEEE Transactions on*, vol. 52, no. 4, pp. 1289–1306, 2006.

[6] A. Septimus and R. Steinberg, "Compressive sensing hardware reconstruction," *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, pp. 3316–3319, 2010.

[7] C. Moler, "magic reconstruction: Compressed sensing," *Mathworks News & Notes*, 2010.

[8] M. Shoaib, N. K. Jha, and N. Verma, "A compressed-domain processor for seizure detection to simultaneously reduce computation and communication energy," in *Custom Integrated Circuits Conference (CICC), 2012 IEEE*. IEEE, 2012, pp. 1–4.

[9] F. Chen, A. P. Chandrakasan, and V. M. Stojanović, "Design and analysis of a hardware-efficient compressed sensing architecture for data compression in wireless sensors," *Solid-State Circuits, IEEE Journal of*, vol. 47, no. 3, pp. 744–756, 2012.

[10] A. Goldberger *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000.

[11] D. Wulsin *et al.*, "Modeling electroencephalography waveforms with semi-supervised deep belief nets: fast classification and anomaly measurement," *Journal of neural engineering*, vol. 8, no. 3, p. 036015, 2011.

[12] A. H. Shoeb and J. V. Guttag, "Application of machine learning to epileptic seizure detection," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 2010, pp. 975–982.

[13] M. Khavari Tavana, A. Kulkarni, A. Rahimi, T. Mohsenin, and H. Homayoun, "Energy-efficient mapping of biomedical applications on domain-specific accelerator under process variation," in *Proceedings of the 2014 International Symposium on Low Power Electronics and Design*, ser. ISLPED '14. New York, NY, USA: ACM, 2014, pp. 275–278. [Online]. Available: http://doi.acm.org/10.1145/2627369.2627654

[14] A. Kulkarni, Y. Pino, M. French, and T. Mohsenin, "Real-time anomaly detection framework for many-core router through machine learning techniques," in *Journal on Emerging Technologies in Computing Systems*, 2016.

[15] A. Kulkarni and T. Mohsenin, "Accelerating compressive sensing reconstruction omp algorithm with cpu, gpu, fpga and domain specific many-core," in *Circuits and Systems (ISCAS), 2015 IEEE International Symposium on*, May 2015, pp. 970–973.

[16] A. M. Kulkarni, H. Homayoun, and T. Mohsenin, "A parallel and reconfigurable architecture for efficient omp compressive sensing reconstruction," in *Proceedings of the 24th Edition of the Great Lakes Symposium on VLSI*, ser. GLSVLSI '14. New York, NY, USA: ACM, 2014, pp. 299–304.