Battery-Less Implantable Continuous EEG Monitoring via Anisotropic Diffusion

Khizar Anjum, *Graduate Student Member, IEEE*, and Dario Pompili, *Fellow, IEEE*

***Abstract*— In this article, we introduce a groundbreaking approach for ultra-low-power hybrid analog-digital processing of multimodal physiological data at multiple locations, emphasizing EEG signals. We propose an innovative analog Convolutional Processing Unit (CvPU) that uniquely harnesses the properties of anisotropic diffusion in electrical circuits for convolution. This novel use of anisotropic diffusion-driven convolution sets our work apart. Additionally, we present a controller architecture that allows for the sequential execution of multiple consecutive convolutional layers using the same CvPU array. The proposed neural network architecture to detect seizures using EEG signals is evaluated on a publicly available clinical dataset. Our CvPU array-based convolution's performance and feasibility metrics have been assessed using SPICE simulation software. Furthermore, we have delved deep into studying the scalability of our approach in terms of power and space and its feasibility for battery-less and implantable applications and have compared it with both digital and hybrid analog-digital methods.**

***Index Terms*— Analog processing circuits, convolutional neural networks, low power electronics, electromagnetic nanonetworks, implantable biomedical devices, body area networks, Internet of Nano Things.**

I. Introduction

The Internet of Things (IoT), especially its nanoscale counterpart, the Internet of Nano Things (IoNT) [2], [3], [4], holds transformative promise for healthcare monitoring. By embedding minuscule sensors within the body, IoNT could offer real-time, granular insights into physiological changes, optimizing early disease detection and personalized treatment regimens. However, actualizing these nanonetworks presents significant obstacles, namely energy consumption and communication [5]. To address these challenges, we propose *in-situ* processing on the nanosensors themselves, bringing down energy requirements by distributing computation and reducing communication costs by not communicating raw physiological signals to microdevices for analysis. Multiple physiological signals can be analyzed to paint a holistic monitoring picture of the body, but not all analyses carry the same difficulty. For example, heart rate or breathing rates are easy to measure and analyze but carry far less information on the body's overall condition than, say, electroencephalogram (EEG) or electrocardiogram (ECG) signals. However, analyzing such complex signals is considerably harder as well. Harnessing the power of neural networks, our approach is geared towards enabling such complex analyses on nanosensors. In this vein, we choose to focus on EEG as a case study for implantable nanosensors, but we acknowledge that the proposed technology can be reconfigured for other complex physiological signals in an on-body implantable IoNT network, leading to comprehensive multi-modal monitoring.

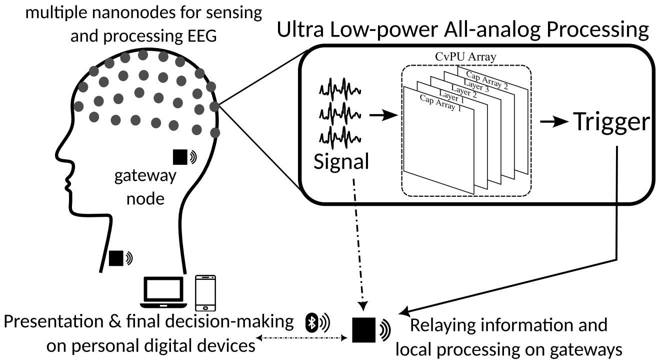

multiple nanonodes for sensing and processing EEG
Ultra Low-power All-analog Processing
CvPU Array
Cap Array 2
Layer 3
Layer 2
Layer 1
Cap Array 1
Signal
Trigger
gateway node
Presentation & final decision-making on personal digital devices
Relaying information and local processing on gateways

Fig. 1. Concept visualization for ultra-low-power hybrid analog-digital processing via multiple nanosensors forming an implantable nanonetwork. EEG data is processed in-situ at multiple nanonodes and subsequent post-processing is done at gateways. The communication between nanodes and gateways is achieved by terahertz band (0.1–10 THz) electromagnetic communication.

A. Motivation

Beyond clinical diagnosis and research, EEG assessments can divulge critical insights into the brain's health and functioning. Long-term and real-time tracking, both in clinical and pre-clinical scenarios, can pinpoint aberrations in cerebral activities, facilitating early detection of neurological maladies like epilepsy and Alzheimer's [6], [7]. While regular clinical EEG sessions may be cumbersome and inaccessible to many, integrating nanosensors, when implanted, can revolutionize this landscape. These implanted nanosensors promise uninterrupted, in-depth cerebral monitoring, bridging the gap between periodic clinical visits. Yet, raw EEG data accumulation alone is insufficient; the transformation of this data into actionable insights is pivotal. Traditionally, this processing is done on bulky digital platforms, culminating in substantial energy

Manuscript received 15 October 2023; revised 25 February 2024; accepted 29 February 2024. Date of publication 9 May 2024; date of current version 18 July 2024. This work was supported by the National Science Foundation (NSF) Real-Time Machine Learning (RTML) under Award 1937403. An earlier version of this paper was presented at the IEEE International Conference on Mobile Ad-Hoc and Smart Systems, September 2023 [DOI: 10.1109/MASS58611.2023.00031]. *(Corresponding author: Khizar Anjum.)*

The authors are with the Department of Electrical and Computer Engineering, Rutgers University, New Brunswick, NJ 08854 USA (e-mail: khizar.anjum@rutgers.edu; pompili@rutgers.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JSAC.2024.3399203.

Digital Object Identifier 10.1109/JSAC.2024.3399203

consumption and delays in procuring analyzed information [8]. Moreover, the operational span of such wearable EEG sensors is short (a few days), and they disrupt users' everyday routines. To counteract these issues, we champion the employment of ultra-low-powered, battery-less processing on an implantable scale. This innovation promises to pare down energy demands related to transmission and computation, ensuring uninterrupted insights into brain health.

B. Our Approach

To combat the drawbacks of current monitoring technology, our proposed approach includes a two-tier wireless sensor network architecture inspired from [2]. The lower tier consists of multiple EEG nanosensors (as shown in Fig. 1) implanted in the head to sense and process EEG signals. The upper tier includes fewer gateways, such as mobile phones or medical devices, that collect data/pre-processed decisions from the nanosensor nodes and forward them to the cloud or personal devices that compile them into meaningful information. Communication between the two tiers can be established through electromagnetic communication in the terahertz frequency band, made possible through nano-antennas based on carbon nanotubes (CNTs) [4]. Furthermore, our conception for continuous EEG monitoring can be extended to multi-modal whole-body monitoring by using multiple low-level sensors doing local processing, as well as multi-modal aggregation on more powerful mobile devices or the cloud (see Fig. 1). By placing multiple sensors on the body and taking advantage of the processing power available in small digital handheld devices, this architecture is both cost-effective and capable of generating a comprehensive picture of an individual's brain health. This can enable early detection of neurological disorders and the development of personalized treatment plans. Leveraging a low-power convolutional framework, our system can offer real-time EEG monitoring without direct reliance on cloud infrastructure. Our approach harnesses an innovative anisotropic diffusion-driven convolutional processor to culminate in an analog detection mechanism. As a final step, the digital component executes computation on a trigger-based mechanism, delivering *just-in-time* determinations to the user.

Our Contributions can be summarized as follows:

- We propose a novel analog Convolutional Processing Unit (CvPU) using the properties of anisotropic diffusion in electrical circuits.
- We propose a novel controller architecture for the sequential execution of multiple consecutive convolutional layers by reusing the CvPU array.
- We propose a novel end-to-end hybrid analog-digital architecture to detect seizures using EEG signals, using hardware-software co-design principles, and evaluate it on a publicly available clinical dataset.
- We evaluate the performance and feasibility metrics of our CvPU array-based convolution using SPICE simulation software.
- We study, in detail, the scalability of our approach, vis-a-vis power as well as space, and compare it with other approaches.

C. Outline

The remainder of this article is organized as follows. In Sect. II, we position our work with respect to works related to our proposal. In Sect. III, we explain our proposed two-step approach. In Sect. IV, we evaluate our approach, first analog and digital domains separately, and then in a hybrid manner. Sect. V involves discussion about the limitations and advantages of our approach. Finally, in Sect. VI, we conclude the article.

II. Related Work

In this section, we thoroughly describe the literature relevant to the Internet of Nano Things (IoNT), EEG signal processing and analysis, wearable EEG devices, and analog computation for neural networks.

A. Internet of Nano Things (IoNT)

The idea of IoNT was described by Akyildiz and Jornet [3], and it fundamentally consists of nanosensors communicating together to form a nanonetwork, for widely varying applications, such as an intrabody nanonetwork for healthcare monitoring or an interconnected office space. Nanosensors are envisioned to communicate at Terahertz frequencies using graphene-based nano-antennas, while a few 'microgateways' are responsible for collecting and parsing data received from these nanosensors, later conveying it to the cloud or other mobile devices for analysis and interpretation. In this context, much of the work done so far has been to investigate ways to communicate between nanosensors reliably [9], [10], [11]. Balasubramaniam and Kangasharju [5] enumerate the challenges faced in realizing the IoNT. The authors put the challenges addressed by our approach—energy efficiency and communication costs—into context, whereby the proposed CvPU architecture specifically addresses sensor-level challenges by reducing bandwidth requirements and middleware computation requirements by introducing intelligence to the nanosensors themselves. Added benefits include continuous, long-term monitoring that has been shown to be more effective at recognizing epilepsy and Alzheimer's early [7].

B. EEG Signal Processing & Analysis

Electroencephalogram (EEG) involves sensing non-invasive electrical signals on the scalp. It has been used for tasks as diverse as motor-imagery inference [12], emotion recognition [13], and speech comprehension [14], as well as preliminary diagnosis for neurobiological conditions such as cognitive impairment, Parkinson's, schizophrenia, and dementia [15]. Furthermore, several types of neural networks have been designed to help with such diagnosis, including but not limited to Convolutional Neural Networks (CNNs) – i.e., EEGNet [16] –, Long Short-Term Memories (LSTMs), as well as Recurrent Neural Networks (RNNs) [17]. Specifically, Lawhern et al. [16] proposed a compact CNN for analyzing EEG signals which surpasses many prominent works in performance metrics. Moreover, SeizureNet [18] proposes an architecture based on several convolutional and dense layers to predict seizure types. However, works such as SeizureNet are designed in isolation, assume unlimited resources, and do not take into account the resources required to process the underlying EEG signals, and hence are unamenable to in-situ processing of EEG signals to alert users of any possible seizure. This work, on the other hand, proposes a novel analog architecture using 1 to 3 orders of magnitude less power than the average digital processor, as well as provides a hybrid analog-digital approach for real-time in-situ processing of EEG signals.

C. Wearable EEG Devices

The wearable devices market has been experiencing significant growth in recent years, driven by advances in technology, increasing demand for health and fitness tracking, and rising consumer awareness of the benefits of wearable devices. Wearable devices include smartwatches, fitness trackers, smart glasses, and other wearable technology that can monitor and track various health and fitness metrics, such as heart rate, steps taken, sleep patterns, and more. In North America, the market for Neurotechnologies is expected to hit USD 38.17 billion, growing at an annual rate of 11.53% [19]. In fact, there are several startups harnessing the power of EEG signals harvested through wearable helmets to target markets as diverse as stress management, mental well-being, as well as sports diagnosis [8]. However, the limiting factors for the state-of-the-art wireless EEG monitoring systems are low battery life (3 to 10 hours) and the dependence on the cloud for timely analysis of the collected data [8]. We note that while this might not be an issue for low-stakes applications, such as entertainment and meditation; for applications such as continuous stress monitoring, real-time concussion monitoring for athletes, or continuous monitoring for seizures, a longer battery life (for longer wearing) as well as a real-time analysis is required. Our proposed solution handles both of these challenges as we propose an in-situ low-powered approach to provide fast as well as low-powered analysis to the end-user, resulting in a tighter feedback loop as well as a longer operational life for EEG monitoring devices.

D. Analog Computation for Neural Networks

There has been a great push toward analog computation for neural networks in recent years. The major motivation behind this push is the ability to break free from the traditional von-Neumann architecture for computing by having computation in-memory [20]. Analog offers a new way out of this by enabling in-memory computation, whereby data can flow through at blazing speeds. The main proposed ways of accomplishing this task have been memristors [21], [22], and resistive processing units (RPUs) [23] based cross-bar arrays. These analog circuits specialize in Matrix-Vector Multiplications (MVMs) or Vector-Matrix Multiplications (VMMs), which form the backbone of the computation in modern neural networks. Simulators for analog design include but are not limited to ALPINE [24], SpiNNaker Project [25], PUMA [26], and DIANA [27]. However, we argue that the impetus behind these projects is more about executing existing digital designs (such as MVMs, or VMMs) using analog circuitry rather than thinking about co-design aspects to produce efficiency in both the algorithm and the hardware. We, on the other hand, propose a co-designed architecture leading to better performance in both aspects. Furthermore, while SPICE models [28], [29], [30] have been developed for simulating large number of memristors, real-life implementations of memristor-based designs are lagging due to the practical considerations, with only small array sizes (256×512, 256×64, etc.) realized in-practice [31]. On the other hand, our proposed architecture, that is entirely based on components amenable to CMOS fabrication and is compliant with Very Large-Scale Integration (VLSI), is realizable and scalable. Furthermore, Correll et al. [32] report that an effective memristor cross-bar implementation requires extensive support circuitry for dealing with the aforementioned problems and ensuring that the computation is performed accurately. An RPU-based implementation [23], [33] solves these issues but lacks the simplicity and the ability to scale efficiently with chip size. Closest analogue to our CvPU design is the Folded Neural Network (FNN) proposed by Hsieh et al. [34], but it is only applicable to fully-connected neural networks, while our proposed design is targeted at Convolutional Neural Networks (CNNs).

III. Proposed Solution

We propose a low-powered approach to process EEG signals in-situ to provide feedback to the user in real time. In order to enable this application, we propose to implement convolution in the analog domain leading to in-situ processing at the nanosensors, which relay their outputs to on-body gateways via electromagnetic THz communication. We describe our proposed approach in the following manner: the isotropic and anisotropic diffusion in electrical systems, then their application in constructing a CvPU, a hardware-software co-designed CNN, and finally we end by describing the approach's integration into an electromagnetic nanonetwork.

A. Isotropic & Anisotropic Diffusion in Analog Circuits

Diffusion is a fundamental process of nature and has been harnessed in many scientific fields, including physics, chemistry, and biology. Our focus, however, is its application in vision. In image processing literature, diffusion refers to a class of algorithms that aim to smooth out the image – similar to how heat pattern diffuses in a 2D material. The equation for diffusion in image processing (as stated in the early vision literature [35]) is typically given by the diffusion equation,

$$\frac{\partial I}{\partial t} = \nabla \cdot (c(x,y,t)\nabla I) = c(x,y,t)\Delta I + \nabla c \cdot \nabla I, \qquad (1)$$

where $I(x,y,t)$ is the image at time t, $c(x,y,t)$ is the diffusion coefficient at input (x,y) and time t, and ∇ is the gradient operator.

In the case of isotropic diffusion, the diffusion coefficient $c(x,y,t)$ is a constant, which means that the diffusion process is equal in all directions. This results in a smoothing effect equivalent to convolving the original image with Gaussian kernel of variance t, i.e., $I(x,y,t) = I_o(x,y) * G(x,y;t)$.

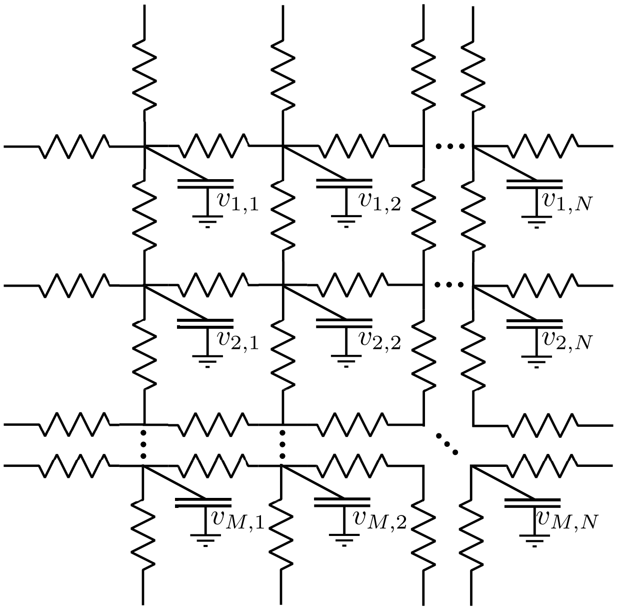


Fig. 2. Circuit implementation for anisotropic diffusion as envisioned by Perona and Malik [35]. This architecture is useful for edge preservation and segmentation but not convolution.

Anisotropic diffusion, on the other hand, is a type of diffusion where the diffusion coefficient varies based on the local image structure, i.e., $c(x, y, t)$ is a function of the local image structure. In this vein, the most relevant implementation is Perona-Malik diffusion [35], where they modified the diffusion coefficient in (1) to be a nonlinear function of the gradient magnitude, i.e.,

$$\frac{\partial I}{\partial t} = \nabla \cdot (g(|\nabla I|)\nabla I) \tag{2}$$

where $|\nabla I|$ represents the gradient magnitude of the image, and $g(|\nabla I|) = c(.)$ is a nonlinear function of the gradient magnitude. The function $g(|\nabla I|)$ is typically defined as:

$$g(|\nabla I|) = \frac{1}{1 + \left(\frac{|\nabla I|}{k}\right)^2} \tag{3}$$

where k is a constant that controls the strength of the diffusion. In practice, this non-linear function $g(.)$ is implemented using a resistive fuse circuit [36], [37], and the process realized using a neat 2D-array as shown in Fig. 2. Each resistor in Fig. 2 is a resistive fuse implementing the non-linear behavior as its I-V curve, as the voltage difference across each resistive fuse corresponds to the gradient of the image. The voltages on the capacitors correspond to the pixel intensities, which are read out after a certain time, depending on the application requirements. However, the simplicity of the original design does not lend itself to anything other than edge preservation using anisotropic diffusion, especially not arbitrary convolution (a linear operation). This is exactly our novelty lies, as we modify the architecture to support a subset of convolutional kernels and then co-design the corresponding algorithm for the analog implementation.

B. Convolutional Processing Unit (CvPU)

Although, the Perona-Malik diffusion [35] circuit (see Fig. 2), was designed for early vision applications, our proposed architecture takes a leap towards general-purpose convolution (although constrained by the number of Degrees of Freedom (DoF)) and is not specific to image pixels, but rather can be applied to any general input array. We take a step further and postulate if there exists a choice of $c(.)$, which is equivalent to convolving input array I with an arbitrary convolutional kernel K, i.e., finding the choices of $c(x, y, t)$, for which (1) is satisfied by the equation (where $*$ denotes the convolution operator),

$$I(x, y, t) = I_o(x, y) * K(x, y; t). \tag{4}$$

1) Mathematical Description of CvPU: It turns out that there does exist a $c(x, y, t)$ which satisfies convolution with an arbitrary kernel K albeit with a modified architecture (shown in Fig. 3) and some restrictions on the degrees of freedom of K. This can be proved by looking at $(i, j)-$th input, corresponding to voltage $v_{i,j}$ as shown in Fig. 3. In our modified architecture, the adjacent inputs in the array are not connected together directly (by a conductance element, see Fig. 2), but rather the summation of adjacent inputs serves as the median between the two – achieved using analogue adder [38]. Furthermore, conductances are defined according to cardinal directions: c_N, c_E, c_S, c_W, c_{NE}, c_{NW}, c_{SE}, and c_{SW} correspond to North, East, South, West, Northeast, Northwest, Southeast, and Southwest respectively. As evident from various layers in Fig. 3, the conductance elements are alternated along their respective directions in the array. As noted in [35], the discrete solution to (1) for a square lattice can be written as a summation of the gradients in each direction. Using $N(i, j) = \{(i-1, j), (i, j+1), (i+1, j), (i, j-1), (i+1, j+1), (i+1, j-1), (i-1, j+1), (i-1, j-1)\}$ to define the set of neighbors to the input (i, j), we can write the evolution of $I_{i,j}$ as,

$$I_{i,j}^{t+1} = I_{i,j}^{t} + \lambda \sum_{n \in N(i,j)} [c_n \cdot \nabla I_n]_{i,j}^{t}, \tag{5}$$

where $I_{i,j}^t$ corresponds to input at node (i, j) which is represented by voltage $v_{i,j}$ in practice. The conductances c are controllable and directly depend on the I-V curves of conductance elements (shown in Fig. 3). The symbols ∇_n denote nearest-neighbor differences between $I_{i,j}$ and $I_n \forall n \in N(i, j)$. Furthermore, the λ parameter controls the variance of convolution (number of passes), the time constant, and the stability of the circuit. For this work, we consider $\lambda \in [0.5, 1]$ for a perfect approximation of the convolutional filter. In Fig. 3, the nodes are arranged so that the neighboring nodes are always equal (at time t) to the sum of the node and the corresponding adjacent input. This is to say that $\forall n \in N(i, j)$:

$$\nabla_n I_{i,j}^t = \lambda(I_n^t + I_{i,j}^t) - I_{i,j}^t \tag{6}$$

Using (6), one can rewrite (5) as:

$$I_{i,j}^{t+1} = \left(1 - \lambda \sum_{n \in N(i,j)} c_n\right) I_{i,j}^t + \lambda \sum_{n \in N(i,j)} c_n I_n^t \tag{7}$$

As $\sum_n c_n$ is a constant for a given choice of conductances, (7) is equivalent to convolution with a 3×3 kernel with 8 Degrees-of-Freedom (DoF). We will now discuss how this convolution is implemented using a 3D circuit design.

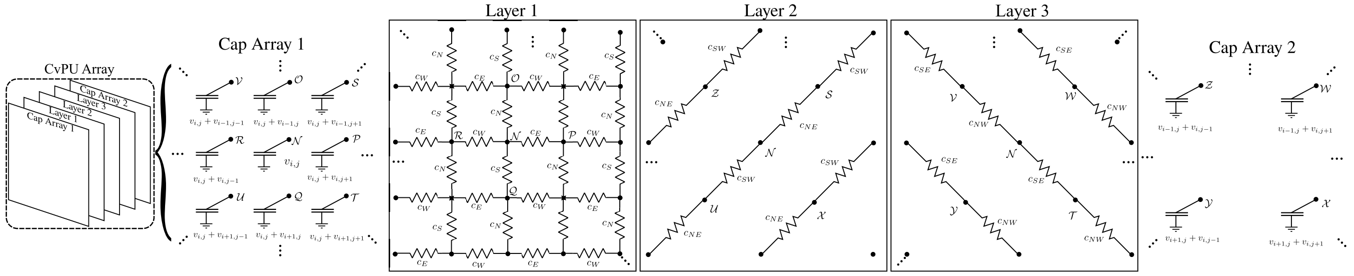


Fig. 3. 3D 8-DoF CvPU design centered on input (i, j), that can be tessellated to create arbitrarily large input arrays for convolution. A 4-DoF CvPU can be implemented by using Cap. Array 1 and Layer 1 only. A total of 13 capacitors (9 in Cap. Array 1 and 4 in Cap. Array 2) are connected to nodes $\mathcal{N}$, $\mathcal{O}$, $\mathcal{P}$, $\mathcal{Q}$, $\mathcal{R}$, $\mathcal{S}$, $\mathcal{T}$, $\mathcal{U}$, $\mathcal{V}$, $\mathcal{W}$, $\mathcal{X}$, $\mathcal{Y}$, and $\mathcal{Z}$. These capacitors are used for input and output, whereby the voltages v written underneath each capacitor are the initial voltages for each node. Through connections to each layer are marked by letters associated with the nodes. Only node $\mathcal{N}$ is connected to all the layers. Furthermore, the nodes marked with a dot $(\cdot)$ are connected to a capacitor element, while the nodes marked with a cross $(\times)$ are left floating. Floating nodes only exist in layer 1. Finally, 8 unique conductance elements (relating to the convolutional kernel) are defined according to spatial directions: c_N, c_E, c_S, c_W, c_{NE}, c_{NW}, c_{SE}, and c_{SW}.

2) 3D Architecture of CvPU: As shown in Fig. 3, the CvPU consists of a circuit with 3D architecture containing through connections between layers. The architecture consists of 2 capacitor arrays (for input and output), and 3 intermediate layers consisting of conductances. The central idea of the architecture is to mediate connections to neighboring inputs to (i, j)-th input by the summation of the two, connected by the relevant conductance. Looking closely at capacitor array 1, we can see that the initial voltages of all the neighboring capacitors are indeed the sum of the principle (i, j)-th voltage and the neighboring voltage. Using through connections to internal layers, they are then connected together by relevant conductances. Finally, capacitor array 2 contains capacitors that are relevant for connections of other inputs, but have to be co-located with other capacitors in array 1. For example, in Fig. 3, node $\mathcal{S}$ mediates connection between $(i, j) - th$ and $(i-1, j+1)$-th input and is placed array 1, but it collides with the placement of node $\mathcal{W}$ that mediates connection between $(i-1, j)$-th and $(i, j+1)$-th inputs. To resolve this, $\mathcal{W}$ is placed in array 2, and the relevant connections of both nodes in the internal layers are shown using the letters.

Furthermore, we would like to mention that the CvPU architecture is designed in a way that it is scalable. The given architecture for a singular node can be extended to an arbitrarily large array for computation. It is also scalable in the other direction, whereby if only capacitor layer 1 and conductance layer 1 are used, a 4-DoF CvPU can be implemented (see [1] for more details). We now discuss the operational parameters of the CvPU and how different kernels can be implemented.

3) Operational Parameters of CvPU: It is desirable that the capacitors at principal nodes $I_{i,j}$ discharge slower than the mediator nodes, whose initial capacitor voltages are the summation of the two neighboring nodes. In our experiments, we have found out that a capacitance ratio of 10 works great, e.g., 10μF for principal nodes, while 100μF for the mediator nodes. Furthermore, (7) shows that the operation of CvPU is equivalent to convolution with the conductance elements $c_n \forall n \in N(i, j)$, therefore during implementation, the conductance values can be chosen accordingly on-the-fly based on the desired kernel to convolve with. The conductance elements can be implemented using a voltage-controlled resistor, i.e., using HRES resistor element [39]. As it is presented, one arbitrarily sized CvPU array can be used to implement one convolutional array in a neural network, followed by further circuit for non-linearities. However, neural networks typically have multiple layers, and in order to save space (limited to small devices), we propose a controller architecture below that can reuse the CvPU array for consecutive neural network layers.

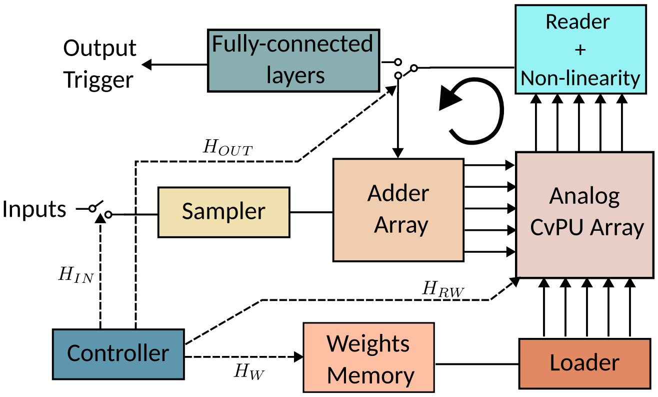


Fig. 4. Controller circuit design for multi-layer computation using limited fixed-sized CvPU array. The controller orchestrates the read/write operations to execute multiple consecutive layers sequentially.

4) Controller Architecture: For a convolutional neural network with L layers, the 1^{st} and L^{th} are considered input and output layers with array size depending on the nature of inputs. In order to reuse the same CvPU array for multiple layers, careful temporal orchestration is needed. The overall concept of our controller circuit is shown in Fig. 4, where consecutive convolutional layers can be executed sequentially from the same neural network. The most basic component of such computation is a single fold. Let the time taken to process one fold be T_F, which also denotes the time-window of EEG signal processed at once, and can involve either single or multi-layer computation (if the CvPU array contains multiple stacked convolutional layers), denoted by n_{lpf} (layers processed per fold). n_{lpf} depends on the realized circuit design and cannot be changed once implemented. One processing cycle—through the whole network—requires n_F folds, resulting in a processing time of $T_P := n_F T_F$. Lastly, the time taken for each fold depends upon the time it takes to charge input- and summation-capacitors in the CvPU array (T_W), the time it takes for diffusion to happen (T_C), and then finally, the time it takes to read the outputs of the array (T_R).

As the weights in the CvPU array are voltage-controlled (using HRES resistor element [39]), we use a capacitor as the

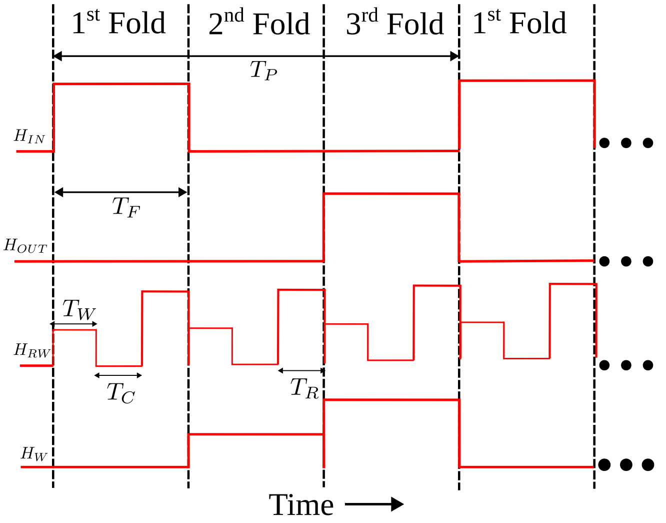


Fig. 5. Example controller output for 3-fold computation ($T_P = 3T_F$) using an arbitrary-sized CvPU array.

basic memory element. We implement a weights-memory that holds the values of weights required for sequential processing. These memories are arrays of capacitors read/written by signals from the controller. The controller uses four signals, namely, H_{IN}, H_{OUT}, H_{RW}, and H_W for executing input, output, write/read, and weight-change operations respectively (see Fig. 4). H_{IN} controls the switch S_{IN}, and is HIGH (connecting to 1^{st} layer output) for the first fold and LOW (connecting to feedback) for all other folds in a processing cycle. H_{OUT} controls the tri-state switch S_{OUT}, and is HIGH (connecting to L^{th} layer input) for the last fold, and LOW (connecting to feedback) for all other folds in a processing cycle. When H_{OUT} is HIGH, the output is forwarded to fully-connected layers to output the final decision by the network. H_{WR} controls read/write-operation to the CvPU array and controls if input is being fed or output is being read from the CvPU array. For both reading and writing, the conductances are made zero so that the capacitors can either be charged (written to) or read from. Finally, H_W changes between n_F discrete levels during the processing window T_P, to load the weights into the loader circuit for sequential computation. A sample output for controller signals is shown in Fig. 5 for $n_F = 3$.

5) Additional Layers: We use analogue adders [38] as a means to implement average pooling in between adjacent convolutional layers in a neural network implemented using the Convolutional Processing Unit (CvPU). In our architecture, we implement both 2×2, and 3×3 average pooling filters in between layers as we focus on the square lattice structure with 4-DoF, and use it for spatio-temporal processing of EEG signals in the next subsection.

C. Hardware-Software Co-Design for EEG Signal Processing

To effectively extract useful information from EEG signals and employ the aforementioned techniques to make real-time decisions based on raw input data, it is important to use a reasonable neural architecture. To this end, Convolutional Neural Networks (CNNs)—such as SeizureNet [18]—have been shown to achieve great performance in classifying EEG signals. Therefore, we choose a CNN architecture for our EEG processing. It is crucial to transform the input signal into a format appropriate for our analog array. For this purpose, we propose to gradually delay the instantaneous analog sensor outputs to provide a window into their temporal behavior, as shown in Fig. 6. For modeling the temporal behavior, convolutional connections work best as they model the relation between future and past time-steps as well as adjacent channels. Mathematically, given an analog output $A(t)$, the corresponding time-delayed signal is given by $A_\tau(t) = A(t - \tau)$. Furthermore, as we have multiple delay elements with delays $\tau_1, \tau_2, \ldots, \tau_M$ with M being the total number of delay-elements, the set of temporal inputs to the convolutional array becomes $\{A(t - \sum_{i=1}^{k} \tau_i) | k = 1, 2, \ldots, M\}$. The spatial resolution, on the other hand, depends on the number of channels we have available at our disposal, denoted by N. All in all, the first input to the analog array is $M \times N$, where M denotes the number of temporal components (proportional to the length of the time-window), and N denotes the number of spatial components. For this work, we choose $M \times N = 300 \times 19$. This can be seen clearly in Fig. 6, where multiple delay-line elements give rise to an array with channels and time as its dimensions. This array is then input into successive 4-DoF convolutional layers with 25, 50, 100, and 150 convolutional filters, respectively. Here, 4-DoF CvPU is used for simplicity in simulation and implementation, but the concept is still valid for 8-DoF CvPU. In the analog domain, the pooling is achieved by simply using analog voltage adders in between the successive convolutional layers, making them equivalent to average pooling. Furthermore, the ReLU non-linearities are implemented using Diode Pair (DP) architecture [33] after each layer. At last, a dense layer (implemented using the Resistive Processing Unit (RPU) [33]) maps the input from the filters into a vector of length 2, to which softmax is applied to determine the final decision from the network.

For training, given that we have a training dataset of input signal windows and labels $(d, y) \in (D, Y)$, where each sample belongs to one of K classes ($Y = 1, 2, \ldots, K$). Our objective is to determine a function $f(D) : D \to Y$ that maps each input d to a label y. To train our model, we use the parameterized function $f(D, \theta)$, where θ are the learned parameters obtained by minimizing the training objective function: $\theta^* = \arg\min_\theta L_{CE}(y, f(D, \theta))$. Here, L_{CE} represents the Cross-Entropy loss, which is applied to the outputs of the ensemble with respect to the ground truth labels. Mathematically, L_{CE} can be expressed as: $L_{CE} = \sum_{k=1}^{K} \mathbf{I}(k = y_i) \log \sigma(O_e, y_i)$, where $O_e = \frac{1}{N_e} \sum_{e=1}^{N} O_k$ denotes the combined logits produced by the ensemble, O_k denotes the logits produced by an individual sub-network, $\mathbf{I}$ is the indicator function, and σ is the SoftMax operation given by: $\sigma(z_i) = \frac{\exp(z_i)}{\sum_{k=1}^{K} \exp(z_k)}$. To initialize the network weights, we use zero-mean Gaussian distributions with a standard deviation of 0.01 and set the biases to 0. At the same time, given the limitations of the nature of analog convolutions, we constrain the 3×3 kernel weights to values allowed by (7). We train the network for 400 epochs with a starting learning rate of 0.001, which is divided by 10 at 50% and 75% of

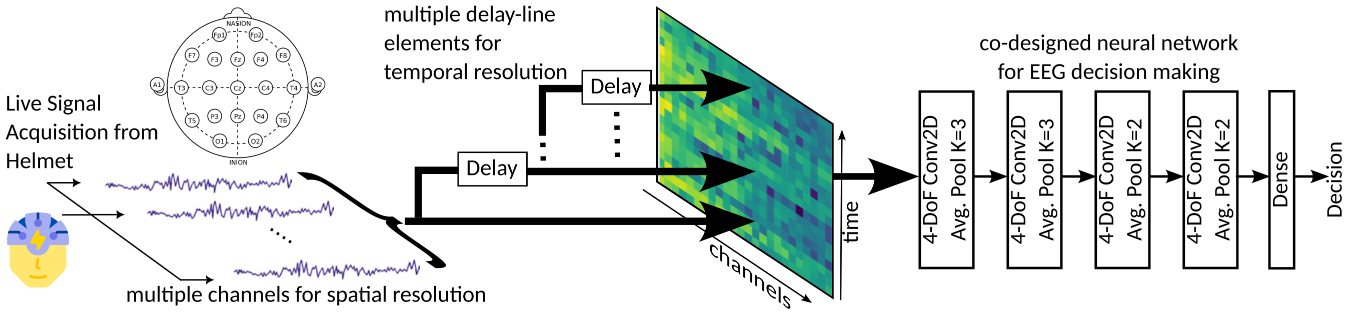

NASION
Fp1
Fp2
F7
F3
Fz
F4
F8
A1
T3
C3
Cz
C4
T4
A2
T5
P3
Pz
P4
T6
O1
O2
INION
Live Signal Acquisition from Helmet
multiple delay-line elements for temporal resolution
Delay
Delay
multiple channels for spatial resolution
time
channels
co-designed neural network for EEG decision making
4-DoF Conv2D Avg. Pool K=3
4-DoF Conv2D Avg. Pool K=3
4-DoF Conv2D Avg. Pool K=2
4-DoF Conv2D Avg. Pool K=2
Dense
Decision

Fig. 6. Our proposed scheme for co-designed analog spatio-temporal processing of EEG signals. The spatial array of signals is gradually delayed to offer insight into the temporal behavior of the instantaneous signal. This spatio-temporal picture is then processed using multiple layers of analog convolutional layers until a decision is reached in analog. An EEG Helmet is shown for validation purposes on available datasets, but the approach is trivially extendable to CvPU-based processing on individual implanted nanosensors.

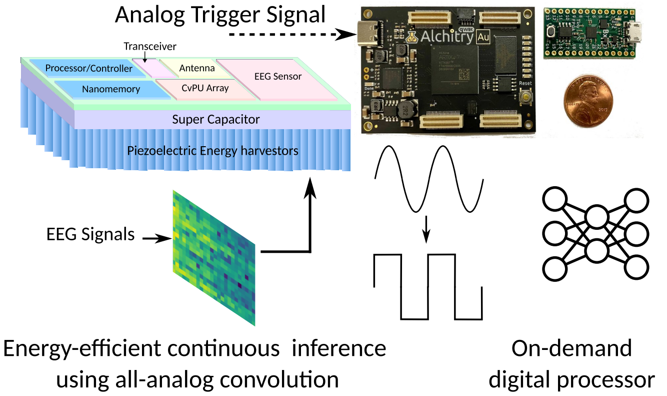


Fig. 7. Hybrid analog-digital architecture for continuous and energy-efficient seizure detection on nanosensors (conceptual design shown on the top left) that can communicate triggers/sensor data to micro-gateways. Here, the analog inference is designed to have a very low False Negative (FN) rate to not miss any probable event, while any False Positives (FP) are further analyzed by the digital system (woken up only when needed, i.e., *just in time*).

the total number of epochs. We also apply a parameter decay of 0.0005 on the weights and biases. Our implementation is based on the PyTorch library [40], and we train the network using the ADAM optimizer with a batch size of 32.

D. Integration Into an Electromagnetic Nanonetwork

Our hybrid analog-digital HW/SW co-designed system revolutionizes EEG seizure detection by leveraging the best of both analog and digital domains. The analog part of the system (CvPU) acts as a pre-stage only, using sensors to collect EEG data and filter out irrelevant information. This stage is energy-efficient and less accurate, but it is crucial in identifying interesting cases that require further analysis. The digital part of our system consists of an FPGA that processes the EEG data in greater detail, providing more accurate results. However, this stage is less energy-efficient. To conserve energy, the digital stage is triggered only when the analog system's output goes high, indicating a positive indication of an EEG seizure. We implement the digital side using Seizure-Net [18] with Multi-Spectral Feature Sampling.

Inspired by the conceptual design presented by Canovas-Carrasco et al. [41], we propose to fit our CvPU array into a nanosensor (shown in Fig. 7), whereby the trigger/sensor data would be communicated to gateways via antennas composed of graphene, i.e., carbon nanotubes (CNTs) [4], using simple modulations such as TS-OOK [42] when an anomaly is detected by an individual nanosensor. Furthermore, a supercapacitor is used as an energy source, coupled with piezoelectric energy harvesting [41] for battery-less operation. Overall, our hybrid analog-digital HW/SW co-designed system provides a comprehensive solution to EEG seizure detection.

IV. Performance Evaluation

In this section, we first introduce the experimental setup used to evaluate the proposed algorithm and then go on to describe the feasibility of analog spatio-temporal processing and end-to-end evaluation of the system. We finally end the section with a scalability, power & noise analysis of the proposed architecture.

A. Experimental Setup

1) EEG Channels Used: There is an internationally recognized system for placing EEG sensors on the scalp, known as the 10-20 system. In this system, electrodes are placed at specific locations on the scalp relative to the landmarks on the skull. The system divides the scalp into regions named according to their position and laterality relative to the midline. The electrodes are labeled with letters and numbers For example, electrodes placed on the midline of the forehead are labeled Fz, while those on the left and right sides of the forehead are labeled F3 and F4, respectively. Similarly, electrodes placed on the left and right sides of the temporal region are labeled T3 and T4, respectively. The system also includes electrodes placed on the mastoid processes behind the ears (M1 and M2) and on the back of the head (O1 and O2) to serve as reference and ground electrodes, respectively. Overall, the placement of EEG sensors follows a standardized system to ensure that recordings can be compared across studies and that results are consistent and reliable. For our experiments, we limit ourselves to a total of 19 channels, including channels FP1, FP2, F3, F4, C3, C4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6, CZ, A1, and A2. The inputs to our experiments were the outputs subtracted from the reference average of all EEG channels.

2) Dataset: The TUH EEG Seizure Corpus (TUH-EEGSC) [43] was utilized as the source of data for our study. It is the largest publicly available dataset of seizure recordings with type annotations worldwide. The dataset was released

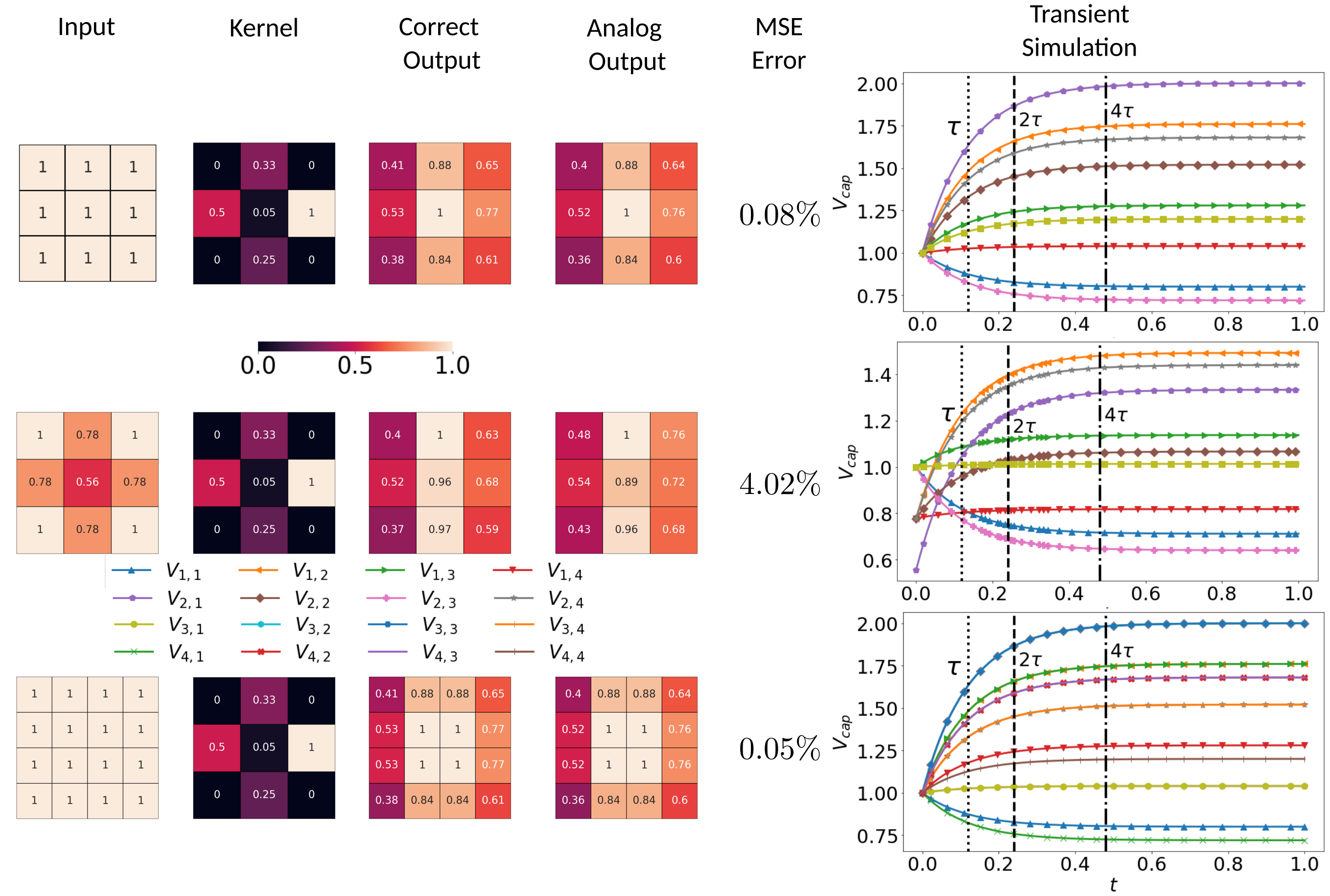


Fig. 8. Simulations of CvPU-based arrays and convolution in SPICE and their corresponding transient plots. All arrays are normalized to be in the range $[0, 1]$, and τ represents the time constant for the arrays (simulated using ngspice).

TABLE I
STATISTICS OF TUH EEG SEIZURE CORP (V2.0.0) IN TERMS OF SEIZURE TYPES, NUMBER OF PATIENTS, AND SEIZURES COUNT

Seizure Type	Patients	Seizures
Focal Non-Specific (FN)	160	649
Generalized Non-Specific (GN)	86	258
Simple Partial Seizure (SP)	2	5
Complex Partial Seizure (CP)	37	166
Absence Seizure (AB)	8	15
Tonic Seizure (TN)	3	10
Tonic Clonic Seizure (TC)	13	19
Myoclonic Seizure (MC)	1	2

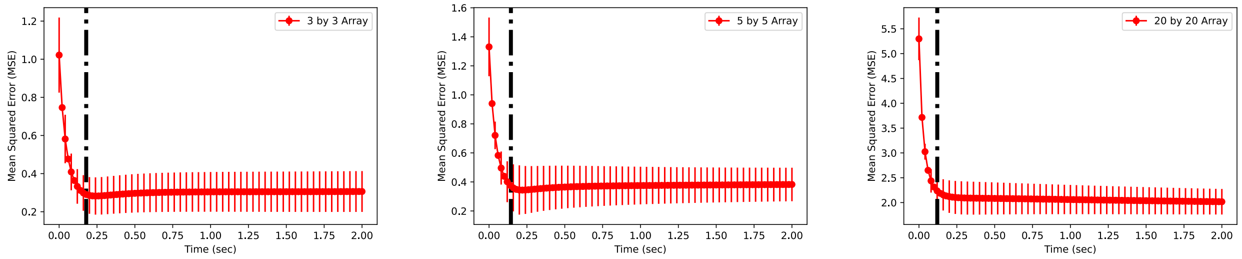
in three versions, with TUH-EEGSC v1.4.0 being released in October 2018, TUH-EEGSC v1.5.2 being released in May 2020, and v2.0.0 being released in March 2022. Only TUH-EEGSC v2.0.0 was available at the time of the analysis and used for results in this article. This corpus has EEG signals that have been manually annotated data for seizure events (start time, stop, channel, and seizure type). Table I provides an overview of TUH-EEGSC's statistics regarding various seizure types and patient numbers. To ensure statistical significance, Myoclonic (MC) seizures were excluded from the study since they had only two seizures, as indicated in Table I. Seizure-level cross-validation was performed for evaluations. For training & evaluation, the dataset was sampled into windows of 300 samples, with a sampling frequency of 250 Hz, with a stride of 100, yielding the input-size of 300 × 19.

3) Simulation Setup: In our study, we conducted digital and analog simulations using different tools and software. For digital simulations, we used Python (version 3.10.10), a widely used programming language for scientific computing and data analysis, for design and execution. For analog simulations, we used two different circuit simulators, LTspice (version XVII) and ngspice (version 36) – both based on the SPICE3 simulator published by the University of California, Berkeley. These tools allowed us to simulate and analyze the behavior of complex analog circuits and validate our designs. To run simulations, we used a Dell Precision 7280 computer, which provided us with the necessary computational power and performance to execute our simulations in a timely and efficient manner.

B. Analog Convolution

In this subsection, we use SPICE circuit simulation software to evaluate and discuss errors in the convolutional calculation, time for steady-state, and second-order effects.

1) SPICE Simulation of Anisotropic Convolution: Furthermore, we simulated 3 × 3 and 4 × 4 anisotropic diffusion arrays in SPICE. These arrays were simulated with capacitors of 50μF, and the conductances c_N, c_E, c_S, and c_W represented



(a) (b) (c)

Fig. 9. Mean Squared Error (MSE) for different sized arrays as simulated in SPICE. We see that the MSE dips until 4τ and then slowly increases as the circuit dissipates the voltage. (a) shows the MSE for a 3 by 3 array; (b) shows the MSE for a 5 by 5 array; (c) shows the MSE for a 20 by 20 array. The results are averaged over 50 random inputs for each experiment (simulated using ngspice).

by resistances of $20k\Omega$, $10k\Omega$, $15k\Omega$, and $5k\Omega$ respectively. Also, note that the same kernel values are shown in Fig. 8 but after normalization of values to be in the range $[0, 1]$. Using the effective parallel resistance that each capacitor sees, and the capacitance (same for all C), we derive the time-constant for the arrays to be $\tau = 0.12$s. Looking at the first row of the input kernel, we see that the analog and correct outputs are very close for the 3×3 array, with mean-squared error as low as 0.08%, and the capacitors' voltages to become steady around 4τ. We see a similar trend for the 4×4 array in the third row, where we see that the error is $< 1\%$ again, and the final voltages show a very smooth trend towards fixed values. For the second row, however, we observe a higher error-rate ($\sim 4\%$). One salient difference here is that the input array is variable, but in spite of higher error, we see that the general structure of the output is preserved. We postulate this is because of the inefficiency of adder circuits, as variable input leads to variations in the linearity of the output. However, the overall trend is clear that anisotropic diffusion-based convolution is possible using circuit elements in SPICE as well.

2) SPICE Simulations for Larger Arrays: In Fig. 9, we use SPICE simulations to explore the Mean Squared Error (MSE) across arrays of varying sizes. The MSE, which measures the gap between expected and actual results, first decreases, hitting its lowest at 4τ, and then gradually rises due to voltage loss in the circuit. 9(a) illustrates the behavior for a 3×3 array, (b) for a 5×5 array, and (c) for a more extensive 20×20 array. It's crucial to highlight that each depicted trend is averaged from 50 random input scenarios to ensure a broad understanding of the MSE patterns.

3) SPICE Simulations for Multi-layer Convolution: In Fig. 10, we examine the Peak Signal to Noise Ratio (PSNR) in relation to varying layer counts for different sizes of CvPU arrays. PSNR is proportional to the log of the inverse of Mean Squared Error (MSE). A swift decline in PSNR is seen when no activation functions are present, challenging the viability of multi-layer analog computations for larger networks. However, with the inclusion of nonlinear activation functions like the Sigmoid, the performance remains stable despite increasing layers. We posit this stability is because a nonlinear activation function, such as Sigmoid, reduces numerical discrepancies between higher and lower output values. This action not only boosts the intermediate PSNR but also prevents these differences from escalating as the signal progresses through subsequent layers. This result is encouraging for analog computation for neural networks as activation functions are necessary for the functioning of a neural network, and they provide a performance boost for our architecture as well.

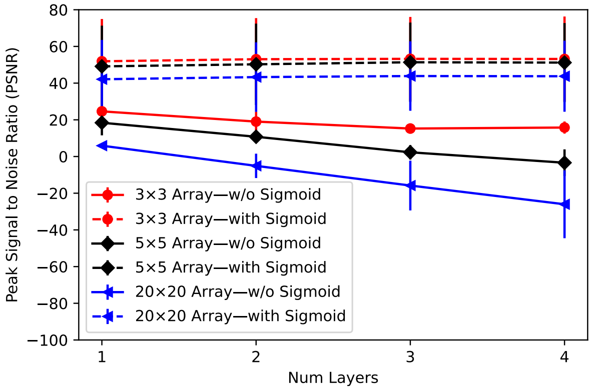
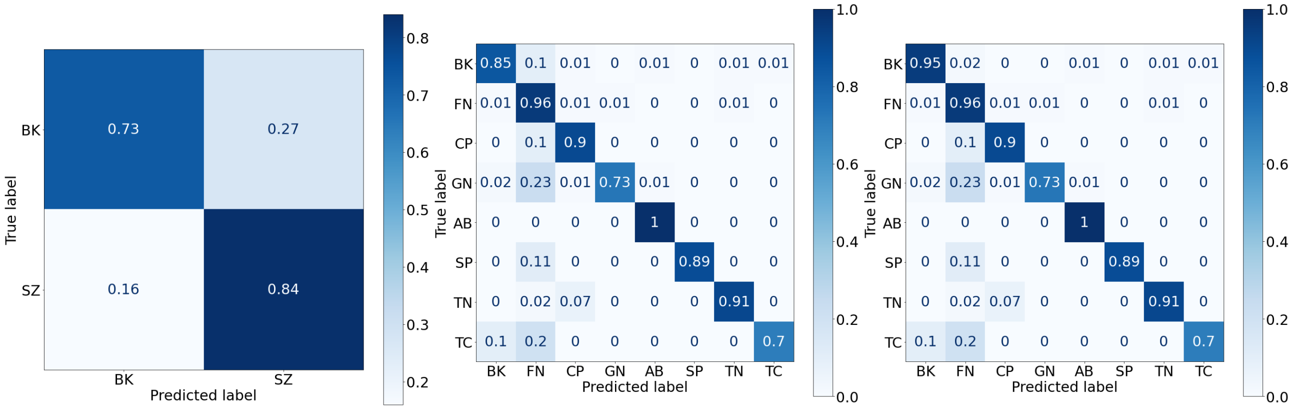


Fig. 10. Peak Signal to Noise Ratio (PSNR) as the number of layers varies for different sizes of CvPU arrays stacked on top of each other. We observe that the PSNR decreases quite fast without activation functions in between, making multi-layer analog computation unfeasible for larger networks. However, with activation functions–e.g., Sigmoid–we get almost no performance drop as the number of layers is increased (simulated using ngspice).

4) Comparison with Related Works: Related works that implement analog computation are geared towards vastly different applications than our proposed CvPU architecture. Architectures such as DIANA [27] and PUMA [26] implement analog convolution using multiple Matrix-Vector Multiplications (MVMs) and have a significant digital part (power hungry). These works are geared towards the acceleration of high-precision convolution at the cost of higher power consumption, while this work is not concerned with high-precision convolution at all, but rather is concerned with power-savings. As a result, the CvPU architecture only consumes power in the order of a few milli-Watts for larger arrays (discussed in IV-D), while architectures such as PUMA [26] consume upwards of 62.5 Watts of power for their computation. Hence, our work is novel in trying to implement convolution with low power, but we pay the price in terms of the loss of generality. Our work is limited in the Degrees-of-Freedom (DoF) of the implemented kernels we can achieve and has the constraint of only being able to implement a 3×3 convolutional kernel.

C. End-to-End Evaluation

In this sub-section, we evaluate the performance of our proposed analog CvPU architecture against other approaches from



(a) (b) (c)

Fig. 11. (a) Confusion matrix for detection using the analog neural network (using CvPU arrays), (b) Confusion matrix for classification using a power-intensive digital network, (c) Performance of the end-to-end joint analog-digital system which uses the analog network as a continuous detector (simulated using python).

the literature, evaluate its efficiency with respect to power, and evaluate metrics such as accuracy and false positives. Fig. 11 shows the confusion matrix for the analog convolutional neural network simulated using anisotropic diffusion-based 4-DoF convolution. Although simulated in python, additive random Gaussian noise is added to the analog convolution to induce a target mean-squared error in it. This is done to make sure that the errors due to convolution are also included in this simulation. However, as shown in Fig. 10, the non-linearities help greatly in the non-deterioration of performance in consecutive layers even in the presence of imperfect analog computation. We see that the network does quite well at detecting the onset of seizures in given windows with an accuracy of 78.5% with a false-negative rate of only 16% (see Fig. 11(a)). Furthermore, the Seizure-Net [18] trained on the TUH seizure corpus attains an accuracy of 86.7% using the resource-intensive digital implementation of the convolutional network. However, as shown in Fig. 11(c), we note that the final system outperforms the digital system as the analog part helps it in filtering out a lot of the data, achieving an accuracy of 88% as a whole. This is not all, as using an energy-efficient analog filter helps reduce power consumption by many orders of magnitude. We can estimate the energy savings by considering that almost 99% of the TUH seizure corpus data is non-seizure and assuming a similar ratio holds in real-life deployments. If the analog detector outputs (correctly or incorrectly) $\sim 73\%$ of the time that the data is non-seizure (background EEG), the digital part is then woken up only 27% of the times, elongating the lifetime of the device by $\sim 4\times$, i.e., from 3 to 10 hours to 12 to 40 hours, providing enough time for a natural charging/recharging schedule, or even being able to harvest the said power from piezoelectric harvestors.

D. Scalability, Power & Noise Analysis

We tackle each of the questions of scalability, power, and noise analyses sequentially in this subsection.

1) Space Scalability: Using the enhanced CvPU architecture as the building block of convolutional layers, we can calculate the number of resistors and capacitors one needs to build a convolutional layer with the input size of $M \times N$. For a 4-DoF 3×3 kernel and a convolutional layer with an output size of $(M+2) \times (N+2)$, the number of resistors comes out to be $(2M+2)(2N+3)+(2N+2)(2M+3)$ while the number of capacitors used comes out to be $(2N+3)(M+2)+(M+1)(N+2)$. Both of these numbers are linear in area (MN), which means that the simple circuit scales linearly as the input size is increased. For an input size of 300×19 (same as our EEG dataset), the number of resistors and capacitors required for one layer comes out to be 48802 and 18703, respectively. Using a 5 nm process, which features around 130–230 million transistors per square millimeter, all the resistors (implemented using 10 transistors) can be furnished in a few μm^2. Furthermore, the above-mentioned silicon process also offers a capacitance density of around 300 $fF/\mu m^2$, meaning that we could implement the full capacitor array using an area of a few mm^2. However, since individual nanosensors do not need full capacitor array to be implemented as they will not be processing all 19 leads of EEG, a smaller array can be furnished in a space of the order of μm^2, making the CvPU feasible for a nanosensor-based application. A few layers of this circuit, hence, may be incorporated into a nano-implantable device that could do in-situ processing and continuous monitoring. We have, unfortunately, not realized a prototype yet and are working actively towards it.

2) Power Scalability: As anisotropic diffusion is essentially a rearrangement of voltage in the capacitors in a particular fashion that mimics convolution, the overall power consumption comes only from the leakage currents in capacitors, currents through resistors during diffusion, and adders. For our simulated 3×3 and 4×4 arrays in SPICE, we observed the current and voltage curves of transient simulations for all individual elements. By multiplying both to get power consumption and then adding for all components, we concluded that the 3×3 and 4×4 arrays consume about 130.8μW and 221.6μW, respectively. An adder only consumes 0.9μW of power, and there are 12 and 24 adders used in 3×3 and 4×4 arrays, respectively, making a total of 10.8μW and 21.6μW respectively. Furthermore, for a 300×19 array (with all 19 EEG channels in conjunction), the power consumption is estimated to be around $55.58 m$W. A small nanosensor, however, does not process all the EEG channels in conjunction, meaning the power

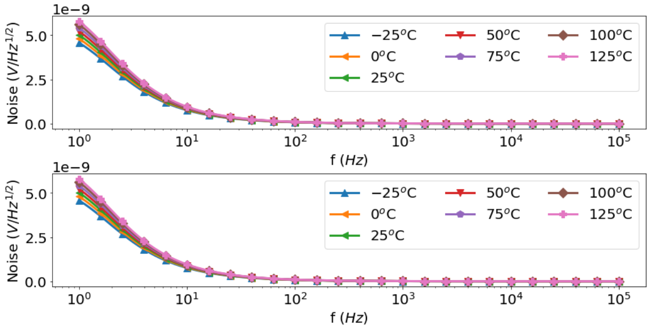


Fig. 12. Noise analysis for 3 × 3 (top) and 4 × 4 (bottom) arrays across temperatures from -25^oC to 125^oC. The noise is comparatively higher (although still in the order of nano-volts per $\sqrt{\text{Hz}}$) at lower frequencies (1-10 Hz). Our operating point for EEG-signal processing is around 250 Hz (simulated using nspice).

consumption will be of the order of a few hundred μW for a small-enough array, and it could be powered using piezoelectric energy harvesters. A detailed model for energy consumption and harvesting, however, is out-of-scope for this article and will be investigated in future studies. On the other hand, efficient FPGAs capable of executing neural networks usually consume from 0.5 (TinyFPGA BX)–10 (Xilinx Ultra96v2) Watts, which is 1 to 3 orders of magnitude greater than the analog power consumption. This is why analog computation is invaluable for energy-efficient continuous monitoring. It is important to note here that the power might also depend further on the capacitance as it dictates how quickly the charge accumulates on the capacitor, dictating how energy-efficient it is (with a lower time-constant). Therefore, more analysis is needed for determining energy consumption with respect to the frequency and the operating point of the circuit.

3) Noise Analysis: We present noise analysis for 3×3 and 4×4 arrays for temperatures ranging from -25^oC to 125^oC, with a step of 25^oC in Fig. 12. The noise is comparatively higher (although still in the order of nano-volts) at lower frequencies (0-10 Hz). However, our operating point for EEG signal processing is around 250 Hz, meaning that the operation of our proposed circuit is resilient against noise introduced from external sources.

V. Discussion

The proposed hybrid analog-digital architecture for EEG processing comes with advantages and limitations. The primary motive behind the hybrid approach is to save energy by using the analog part (implemented using CvPU) to act as a pre-filter to the digital part, which can then wake up and expend energy in detecting and identifying the seizure type. By using novel CvPU as the analog part of the implementation, one inherits its limitations, such as its inability to represent a general convolutional kernel, but only a kernel with 8-DoF at the maximum. Furthermore, the values of the resistances and the capacitances limit its speed, as one has to wait atleast until the time constant of the circuit for stable readings. There is also the limitation concerning the ability to only implement a 3×3 kernel. However, even with all these limitations, one has to keep in mind that the analog part is designed to be a filter with a low false-negative rate so that it can filter out normal signals but not miss any positive events. After the digital part is woken up, the architecture can be made much more efficient and complex for proper computation, leading to the proper identification of seizure events relating to EEG. In return for the limitations of our architecture, the benefit we achieve is low-powered computation with an architecture that is small enough to be implanted. Indeed, as shown in the Sect. IV-D, the considerations of space and power show that the resulting device (comprising CvPU) can be small enough—a few μm^2—to be implantable in the human body and allow for battery-less monitoring (at least for the analog part), while the digital gateways will be needed to be powered.

VI. Conclusion

The article proposes a novel neural network architecture for the early detection of seizures using EEG signals and presents its evaluation on a clinical dataset. An analog CvPU architecture implementation is proposed for continuous in-situ processing, with estimated energy consumption compared to digital and hybrid analog-digital approaches. Our system offers real-time EEG monitoring without the direct need for cloud infrastructure, ensuring extended observation intervals without the hassle of frequent battery replacements or recharges. The culmination of our efforts is a hybrid system with an analog detection mechanism that simulates the time-sequential nature of EEG signals. The digital component executes computations based on a trigger mechanism, delivering timely determinations to users. A SPICE-based evaluation is conducted, and noise analysis is performed at various temperatures. We show, through extensive performance evaluation, that the proposed system can be used for wearable and implantable applications for monitoring physiological signals.

References

[1] K. Anjum and D. Pompili, "Anisotropic diffusion-based analog CNN architecture for continuous EEG monitoring," in *Proc. IEEE 20th Int. Conf. Mobile Ad Hoc Smart Syst. (MASS)*, Sep. 2023, pp. 1–9.

[2] I. F. Akyildiz, F. Brunetti, and C. Blázquez, "Nanonetworks: A new communication paradigm," *Comput. Netw.*, vol. 52, no. 12, pp. 2260–2279, Aug. 2008.

[3] I. F. Akyildiz and J. M. Jornet, "The Internet of Nano-Things," *IEEE Wireless Commun.*, vol. 17, no. 6, pp. 58–63, Dec. 2010.

[4] J. M. Jornet and I. F. Akyildiz, "The Internet of Multimedia Nano-Things," *Nano Commun. Netw.*, vol. 3, no. 4, pp. 242–251, Dec. 2012.

[5] S. Balasubramaniam and J. Kangasharju, "Realizing the Internet of Nano Things: Challenges, solutions, and applications," *Computer*, vol. 46, no. 2, pp. 62–68, Feb. 2013.

[6] B. Cretin et al., "Do we know how to diagnose epilepsy early in Alzheimer's disease?" *Revue Neurologique*, vol. 173, no. 6, pp. 374–380, Jun. 2017.

[7] A. Horváth, A. Szűcs, G. Barcs, J. L. Noebels, and A. Kamondi, "Epileptic seizures in Alzheimer disease," *Alzheimer Disease Associated Disorders*, vol. 30, no. 2, pp. 186–192, Apr. 2016.

[8] C. He et al., "Diversity and suitability of the state-of-the-art wearable and wireless EEG systems review," *IEEE J. Biomed. Health Informat.*, vol. 27, no. 8, pp. 3830–3843, Aug. 2023.

[9] D. Bi, A. Almpanis, A. Noel, Y. Deng, and R. Schober, "A survey of molecular communication in cell biology: Establishing a new hierarchy for interdisciplinary applications," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1494–1545, 3rd Quart., 2021.

[10] H. Elayan, O. Amin, B. Shihada, R. M. Shubair, and M.-S. Alouini, "Terahertz band: The last piece of RF spectrum puzzle for communication systems," *IEEE Open J. Commun. Soc.*, vol. 1, pp. 1–32, 2020.

[11] I. F. Akyildiz, C. Han, Z. Hu, S. Nie, and J. M. Jornet, "Terahertz band communication: An old problem revisited and research directions for the next decade," *IEEE Trans. Commun.*, vol. 70, no. 6, pp. 4250–4285, Jun. 2022.

[12] A. Al-Saegh, S. A. Dawwd, and J. M. Abdul-Jabbar, "Deep learning for motor imagery EEG-based classification: A review," *Biomed. Signal Process. Control*, vol. 63, Jan. 2021, Art. no. 102172.

[13] D. O. Bos, *EEG-Based Emotion Recognition—The Influence of Visual and Auditory Stimuli*. Naarden, The Netherlands: Capita Selecta (MSc Course), 2006, pp. 1–17.

[14] M. Gillis, J. Vanthornhout, J. Z. Simon, T. Francart, and C. Brodbeck, "Neural markers of speech comprehension: Measuring EEG tracking of linguistic speech representations, controlling the speech acoustics," *J. Neurosci.*, vol. 41, no. 50, pp. 10316–10329, Dec. 2021.

[15] A. M. Alvi, S. Siuly, and H. Wang, "Neurological abnormality detection from electroencephalography data: A review," *Artif. Intell. Rev.*, vol. 55, no. 3, pp. 2275–2312, Mar. 2022.

[16] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, "EEGNet: A compact convolutional neural network for EEG-based brain–computer interfaces," *J. Neural Eng.*, vol. 15, no. 5, Oct. 2018, Art. no. 056013.

[17] K. M. Hossain et al., *Status of Deep Learning for EEG-Based Brain–Computer Interface Applications*. Baltimore, MD, USA: UMBC Student Collection, 2023.

[18] U. Asif, S. Roy, J. Tang, and S. Harrer, "SeizureNet: Multi-spectral deep feature learning for seizure type classification," in *Proc. Mach. Learn. Clin. Neuroimaging Radiogenomics Neuro-Oncol.*, Lima, Peru, Oct. 2020, pp. 77–87.

[19] (2023). *Neurotechnology Market Global Industry Analysis, Size, Share, Growth, Trends, Regional Outlook, and Forecast 2023-2032*. Accessed: Apr. 6, 2023. [Online]. Available: https://www.precedenceresearch.com/neurotechnology-market

[20] A. Sebastian, M. Le Gallo, R. Khaddam-Aljameh, and E. Eleftheriou, "Memory devices and applications for in-memory computing," *Nature Nanotechnol.*, vol. 15, no. 7, pp. 529–544, Jul. 2020.

[21] L. Chua, "Memristor-the missing circuit element," *IEEE Trans. Circuit Theory*, vol. CT-18, no. 5, pp. 507–519, Sep. 1971.

[22] M. D. Ventra, Y. V. Pershin, and L. O. Chua, "Circuit elements with memory: Memristors, memcapacitors, and meminductors," *Proc. IEEE*, vol. 97, no. 10, pp. 1717–1724, Oct. 2009.

[23] H. Assaf, Y. Savaria, and M. Sawan, "Vector matrix multiplication using crossbar arrays: A comparative analysis," in *Proc. 25th IEEE Int. Conf. Electron., Circuits Syst. (ICECS)*, Dec. 2018, pp. 609–612.

[24] J. Klein et al., "ALPINE: Analog in-memory acceleration with tight processor integration for deep learning," *IEEE Trans. Comput.*, vol. 72, no. 7, pp. 1985–1998, Jul. 2023.

[25] S. B. Furber, F. Galluppi, S. Temple, and L. A. Plana, "The spinnaker project," *Proc. IEEE*, vol. 102, no. 5, pp. 652–665, May 2014.

[26] A. Ankit et al., "PUMA: A programmable ultra-efficient memristor-based accelerator for machine learning inference," in *Proc. 24th Int. Conf. Architectural Support Program. Lang. Operating Syst.*, Apr. 2019, pp. 715–731.

[27] P. Houshmand et al., "DIANA: An end-to-end hybrid digital and analog neural network SoC for the edge," *IEEE J. Solid-State Circuits*, vol. 58, no. 1, pp. 203–215, Jan. 2023.

[28] B. Li and G. Shi, "A native SPICE implementation of memristor models for simulation of neuromorphic analog signal processing circuits," *ACM Trans. Des. Autom. Electron. Syst.*, vol. 27, no. 1, pp. 1–24, Jan. 2022.

[29] K. Adam, K. Smagulova, and A. James, "Generalised analog LSTMs recurrent modules for neural computing," *Frontiers Comput. Neurosci.*, vol. 15, Sep. 2021, Art. no. 705050.

[30] S. Kvatinsky, M. Ramadan, E. G. Friedman, and A. Kolodny, "Vteam: A general model for voltage-controlled memristors," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 62, no. 8, pp. 786–790, Aug. 2015.

[31] J. M. Correll et al., "An 8-bit 20.7 TOPS/W multi-level cell ReRAM-based compute engine," in *Proc. IEEE Symp. VLSI Technol. Circuits*, Jun. 2022, pp. 264–265.

[32] J. M. Correll et al., "Analog computation with RRAM and supporting circuits," in *Analog Circuits for Machine Learning, Current/Voltage/Temperature Sensors, and High-Speed Communication: Advances in Analog Circuit Design 2021*. Cham, Switzerland: Springer, 2022, pp. 17–32.

[33] Y.-T. Hsieh, K. Anjum, S. Huang, I. Kulkarni, and D. Pompili, "Neural network design via voltage-based resistive processing unit and diode activation function—A new architecture," in *Proc. IEEE Int. Midwest Symp. Circuits Syst. (MWSCAS)*, Aug. 2021, pp. 59–62.

[34] Y.-T. Hsieh, K. Anjum, and D. Pompili, "Ultra-low power analog folded neural network for cardiovascular health monitoring," *IEEE J. Biomed. Health Informat.*, early access, Mar. 14, 2024, doi: 10.1109/JBHI.2024.3375762.

[35] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 7, pp. 629–639, Jul. 1990.

[36] J. G. Harris, C. Koch, and J. Luo, "A two-dimensional analog VLSI circuit for detecting discontinuities in early vision," *Science*, vol. 248, no. 4960, pp. 1209–1211, Jun. 1990.

[37] P. C. Yu, S. J. Decker, H.-S. Lee, C. G. Sodini, and J. L. Wyatt, "CMOS resistive fuses for image smoothing and segmentation," *IEEE J. Solid-State Circuits*, vol. 27, no. 4, pp. 545–553, Apr. 1992.

[38] H. Chaoui, "CMOS analogue adder," *Electron. Lett.*, vol. 31, no. 3, pp. 180–181, Feb. 1995.

[39] C. Mead, *Analog VLSI and Neural Systems*. Reading, MA, USA: Addison-Wesley, 1989.

[40] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," in *Proc. NIPS*, Dec. 2019, pp. 8024–8035.

[41] S. Canovas-Carrasco, A.-J. Garcia-Sanchez, F. Garcia-Sanchez, and J. Garcia-Haro, "Conceptual design of a nano-networking device," *Sensors*, vol. 16, no. 12, p. 2104, Dec. 2016.

[42] J. M. Jornet and I. F. Akyildiz, "Femtosecond-long pulse-based modulation for terahertz band communication in nanonetworks," *IEEE Trans. Commun.*, vol. 62, no. 5, pp. 1742–1754, May 2014.

[43] V. Shah et al., "The temple university hospital seizure detection corpus," *Frontiers Neuroinform.*, vol. 12, p. 83, Nov. 2018.

Khizar Anjum (Graduate Student Member, IEEE) received the B.S. degree from Lahore University of Management Sciences (LUMS) in 2019 and the M.S. degree from Rutgers University in 2022, where he is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering (ECE), under the supervision of Dr. Pompili. His Ph.D. research is focused on enabling deep-learning applications on resource-constrained systems such as drones, digital sensors, and robots. He was a Research Assistant with LUMS, where he worked on utilizing deep neural networks for early diagnosis of Parkinson's disease, a project mixing advanced signal-processing techniques with medical knowledge and neural networks. He is the co-inventor of multiple filed and provisional patents. He has published in top-tier IEEE and ACM conferences, including IEEE PerCom, IEEE MASS, IEEE UComms, IEEE MWSCAS, and ACM WUWNet. His research interests include the intersection of approximate computing, HW-SW co-design, artificial intelligence, and signal processing. He has received several awards, including multiple prestigious ACM and IEEE conference travel grants, the Teaching Assistant of the Year Award at Rutgers University in 2020, and four consecutive Dean's Honor List Awards at LUMS from 2016 to 2019. He was awarded an NMF Gold Medal for Outstanding Performance. He is a reviewer for several top-tier IEEE journals, including IEEE TRANSACTIONS ON MOBILE COMPUTING and IEEE TRANSACTIONS ON ROBOTICS.

Dario Pompili (Fellow, IEEE) received the Laurea (combined B.S. and M.S.) and Ph.D. degrees in telecommunications and system engineering from the University of Rome "La Sapienza," Italy, in 2001 and 2004, respectively, and the Ph.D. degree in ECE from Georgia Institute of Technology in 2007. Since joining Rutgers University in 2007, he has been the Director of the CPS Laboratory, which focuses on next-generation radio access networks, mobile edge computing, wireless communications and networking, acoustic communications, and sensor networks. He is currently a Professor with the Department of ECE, Rutgers University. He has published about 200 refereed scholar publications, some of which received best paper awards with more than 16K citations. He has an H-index of 50 and an i10-index of 138 (Google Scholar, February 2024). He was elevated to a fellow of the IEEE Communications Society in 2021. Since 2019, he has been a Distinguished Scientist of ACM. He has received several prestigious awards in his career, including the NSF CAREER'11, the ONR Young Investigator Program'12, and the DARPA Young Faculty'12 Award. In 2015, he was nominated as the Rutgers-New Brunswick Chancellor's Scholar. He served on many international conference committees taking on various leading roles. He is currently serving as an Associate Editor for IEEE TRANSACTIONS ON MOBILE COMPUTING and the Area Chair for IEEE INFOCOM.