



OPEN ACCESS

EDITED BY

Yang Zhan,
Shenzhen Institutes of Advanced Technology
(CAS), China

REVIEWED BY

Zhenghui Gu,
South China University of Technology, China
Hossein Hosseini-Nejad,
K. N. Toosi University of Technology, Iran

*CORRESPONDENCE

Jie Yang
✉ yangjie@westlake.edu.cn
Mohamad Sawan
✉ sawan@westlake.edu.cn

SPECIALTY SECTION

This article was submitted to
Neuroprosthetics,
a section of the journal
Frontiers in Neuroscience

RECEIVED 09 November 2022

ACCEPTED 03 January 2023

PUBLISHED 23 January 2023

CITATION

Tian F, Yang J, Zhao S and Sawan M (2023)
NeuroCARE: A generic neuromorphic edge
computing framework for healthcare
applications.
Front. Neurosci. 17:1093865.
doi: 10.3389/fnins.2023.1093865

COPYRIGHT

© 2023 Tian, Yang, Zhao and Sawan. This is an
open-access article distributed under the terms
of the [Creative Commons Attribution License
\(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The use, distribution or reproduction in
other forums is permitted, provided the original
author(s) and the copyright owner(s) are
credited and that the original publication in this
journal is cited, in accordance with accepted
academic practice. No use, distribution or
reproduction is permitted which does not
comply with these terms.

NeuroCARE: A generic neuromorphic edge computing framework for healthcare applications

Fengshi Tian^{1,2}, Jie Yang^{1*}, Shiqi Zhao¹ and Mohamad Sawan^{1*}

¹CenBRAIN Neurotech, School of Engineering, Westlake University, Hangzhou, Zhejiang, China, ²The Hong Kong University of Science and Technology (HKUST), New Territories, Hong Kong SAR, China

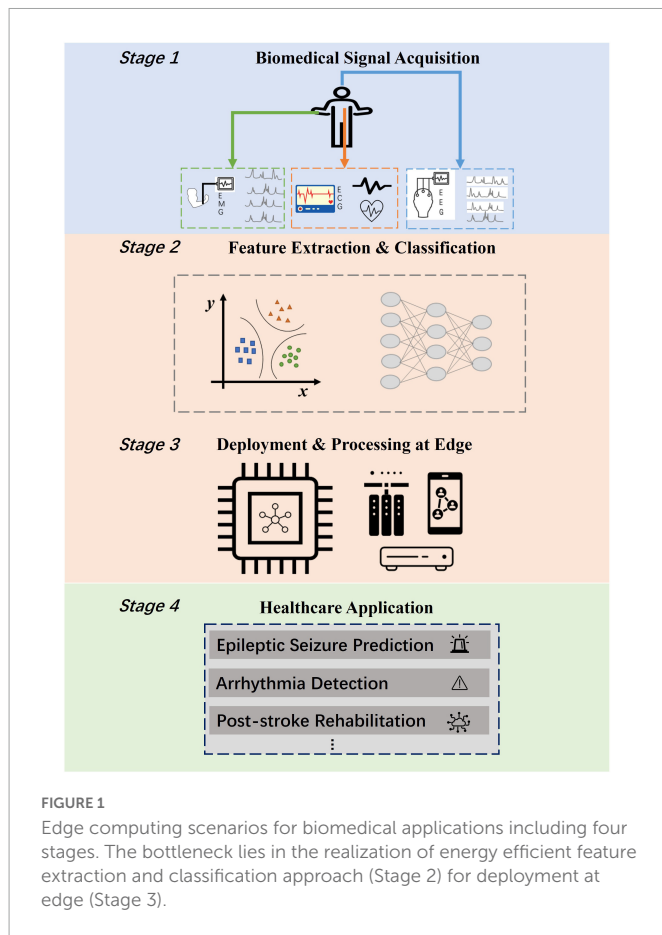
Highly accurate classification methods for multi-task biomedical signal processing are reported, including neural networks. However, reported works are computationally expensive and power-hungry. Such bottlenecks make it hard to deploy existing approaches on edge platforms such as mobile and wearable devices. Gaining motivation from the good performance and high energy-efficiency of spiking neural networks (SNNs), a generic neuromorphic framework for edge healthcare and biomedical applications are proposed and evaluated on various tasks, including electroencephalography (EEG) based epileptic seizure prediction, electrocardiography (ECG) based arrhythmia detection, and electromyography (EMG) based hand gesture recognition. This approach, NeuroCARE, uses a unique sparse spike encoder to generate spike sequences from raw biomedical signals and makes classifications using the spike-based computing engine that combines the advantages of both CNN and SNN. An adaptive weight mapping method specifically co-designed with the spike encoder can efficiently convert CNN to SNN without performance deterioration. The evaluation results show that the overall performance, including the classification accuracy, sensitivity and F1 score, achieve 92.7, 96.7, and 85.7% for seizure prediction, arrhythmia detection and hand gesture recognition, respectively. In comparison with CNN topologies, the computation complexity is reduced by over 80.7% while the energy consumption and area occupation are reduced by over 80% and over 64.8%, respectively, indicating that the proposed neuromorphic computing approach is energy and area efficient and of high precision, which paves the way for deployment at edge platforms.

KEYWORDS

epileptic seizure prediction, arrhythmia detection, hand gesture recognition, biomedical signal processing, neuromorphic computing

1. Introduction

The various biomedical datasets have provided the possibilities for the realization of a wide spectrum of medical applications. Electroencephalography (EEG) signals could be utilized for predicting epileptic seizure which is recognized as one of the most severe neurological diseases (Mirowski et al., 2009), motion recognition (Abiri et al., 2019), and stroke risk prediction which is ranked as the first leading causes of death and disability worldwide (Chen and Sawan, 2021).



Electromyography (EMG) signals could pave the way for hand gesture recognition systems which have been proven to be immensely helpful in many human-computer interactive scenarios like treatment for post-stroke rehabilitation (Chen et al., 2019), and active prosthesis control (Parajuli et al., 2019). Electrocardiography (ECG) signals provide potentials for remote arrhythmia detection, which is one major cause of death nowadays according to World Health Organization (2017). The healthcare applications mentioned above always require at edge deployment for the convenience of patients and the enhancement of physician's diagnosis. However, the computation and memory resources of edge devices are extremely limited. Figure 1 depicts at-edge system scenarios for various healthcare applications. Such systems consist of four common stages: biomedical signal acquisition stage, signal feature extraction and classification approach stage, edge platform deployment stage and application realization stage. To empower the realization of such systems, it is of great significance to propose a high-performance, energy-efficient, and generic feature extraction and classification approach for edge healthcare.

Recently, several AI-based algorithms have been used for various kinds of biomedical signals analysis tasks, like convolutional neural network (CNN) and recurrent neural network (RNN). A Short-Time-Fourier-Transform (STFT) based 2D-CNN was proposed for EEG signal processing by Truong et al. (2018a). Eberlein et al. (2018) designed a CNN with deep layers for seizure onset prediction on the time domain utilizing raw EEG signals. Xu et al. (2020) designed channel-wise convolution kernels to deal with seizure prediction and achieved a sensitivity of 98.2%. Atzori et al. (2016) and Wei et al. (2019) both applied CNN method to deal with EMG data.

Hu et al. (2018) used a hybrid recurrent-convolution network to extract the sequential features of EMG signals, while Simão combined gated recurrent unit (GRU) and long-short term memory (LSTM) with the same aim (Simão et al., 2019). Kiranyaz et al. (2016) used 1D CNN to classify two major phases of ECG for arrhythmia detection and achieved sensitivity of 93.9%. The above methods all show advantages in classification accuracy. However, they all require tremendous amounts of computation resources, which proves to be a challenging task for edge platforms such as wearable and implantable devices. Recently, researchers have explored hardware friendly approaches for feature extraction and classification. Truong et al. (2018b) studied on deploying integer and binary weights in CNN, managing to achieve an over seven times reduction on weight size. Zhao et al. (2020) proposed a single dimension CNN with binary weights to deal with EEG data, which reduces the required memory for parameters by 86.12%. However, Truong et al. (2018b) still utilized integer weights of 4-bit width in their CNN and Zhao et al. (2020) still used fully precise values as dataflow in several layers of the neural network, which waste both energy consumption and computation resources. In conclusion, the existing approaches still face the drawback of balancing task performance with computation complexity. It remains as a problem to reduce data amount as much as possible while keeping the performance at a high level for edge biomedical applications.

Biologically inspired, spiking neural networks (SNNs), which stand as the most demonstrative form of neuromorphic computing approaches nowadays, have shown advantages in energy efficiency and performance in classification tasks (Kasabov, 2014; Srinivasan et al., 2016; Roy et al., 2019) and are expected to be the next generation of AI (Ghosh-dastidar and Adeli, 2009). To detect epilepsy seizure onsets based on EEG, Guo et al. (2017) designed an SNN utilizing supervised training methods and achieved an accuracy of 92.67%. Ma et al. (2020) designed an SNN based on reservoir computing for EMG classification. Amirshahi and Hashemi (2019) proposed an SNN for cardiac monitoring and achieved a sensitivity of 80.2%. Therefore, SNNs can provide a possible solution for deploying energy-efficient edge healthcare applications. Conventional unsupervised training methods like spike-timing-dependent plasticity (STDP) (Pu and Cook, 2015), Tempotron (Iyer and Chua, 2020), and SpikeProp (Shrestha and Song, 2018) are commonly utilized in existing SNNs (Pu and Cook, 2015; Amirshahi and Hashemi, 2019; Ma et al., 2020). However, it remains a problem to design and train spike-based neural networks as deep as CNN based on these methods (Tavanaei et al., 2019), which leads to limitations in both the performance and the applications of SNN. Although some work (Lee et al., 2018) try to improve STDP scheme for training multi-layer SNN, the performance is still insufficient to handle complicated signal analysis tasks. Combining the advantages of both CNN and SNN, Spiking CNNs are proposed to solve the drawbacks described above (Rueckauer et al., 2017). The weights of a Spiking-CNN are gained in CNN shadow training *via* backpropagation. The trained weights are then restored and correspondingly mapped onto the designed SNN. A unique spike encoder is required in spike-based computing methods to generate time-based spike sequences from the input data, which is specifically adapted to process various types of signals and tasks. In this way, Spiking-CNN succeeds to achieve high classification performance while much reduce the required computation resources compared to CNN. Diehl et al. (2015) did some primary work in the exploration of Spiking-CNN and conducted evaluation on MNIST dataset. Cao et al. (2015) also

explored the potentials of Spiking-CNN in object recognition tasks. However, the effect of the spike encoder is overlooked in their work. For conversion based SNNs dealing with classification tasks, the key of keeping high performance is to rebuild the relative statistical data distribution of feature maps between layers in the format of spike sequences. The spike sequences are generated *via* spike encoder, which directly affects the data distribution of features.

To solve the existing drawbacks of unbalance among performance, versatility, and computation efficiency, we propose the implementation of NeuroCARE, a generic neuromorphic computing approach which is adaptive for various edge healthcare and biomedical applications. NeuroCARE first converts raw biologic signals to spikes through the proposed sparse Gaussian spike encoder, which rebuilds the relative statistical data distribution of feature maps in time domain. The dedicated channel-wise network structure of NeuroCARE manages to extract spatial-temporal features and stands as the foundation of accurate and efficient processing. The adaptive weight mapping scheme in NeuroCARE is codesigned with the spike encoder to achieve high performance. Moreover, NeuroCARE can be implemented on hardware in a totally multiplier free fashion which reduces both the hardware cost and power consumption significantly.

In our experiment, NeuroCARE reduces the computation complexity by over 80.7%, and the energy consumption and area occupation are reduced by over 80% and over 64.8%, respectively, showing advantages when compared to other systems with CNN topology.

The organization of this paper is as following. Section “2 Module and architecture designs” describes the proposed system architecture and design flow, where detailed modules of NeuroCARE are shown. Implementations, experiments, and relevant results are summarized in Section “3 Experiments and evaluation results,” respectively. Section “4 Conclusion” concludes this paper.

2. Module and architecture designs

The processing strategy of NeuroCARE consists of two stages (Figure 2). The shadow training phase aims to get weights *via* backpropagation in the designed CNN topology (Figure 2A), which then transforms into neuromorphic SNN fashion (Figure 2B). The trained weights are first adapted and then mapped onto the Spiking-CNN *via* the proposed adaptive weight mapping method (Figure 2C) correspondingly. During the inference phase, a designed sparse Gaussian encoder is utilized to transform the biomedical signals to encoded spike sequences on the time domain. Then the neuromorphic engine operates inference to realize the classifications.

2.1. Sparse Gaussian spike encoder

In this study, we propose a sparse Gaussian spike encoding algorithm to generate time-dependent spike sequences with adaptive sparsity from the input continuous biomedical signals into and thus discrete the input data features in time domain.

Unlike other types of spike encoders such as (Donati et al., 2019) and (Davidson and Furber, 2021) where the encoder can only be applied to specific tasks, the proposed method can achieve better versatility and scalability with the codesign of the proposed

weight mapping scheme. Such co-design scheme also ensures that the proposed method has the potential to apply on various network structure and various tasks. Relevant mathematical deduction will be presented in Part E of Section “2 Module and architecture designs.” Another priority of the proposed encoder lies in that it has better sparsity *via* mimicking the randomness of biological activities. Compared with prior rate-encoding and delta-encoding scheme, the proposed spike encoder could rebuild feature distributions in the time domain using far fewer time steps, thus bringing more efficiency. Moreover, the proposed spike encoder can be implemented in pure digital design and has good reconfigurability. Part F of Section “3 Experiments and evaluation results” presents the detail of circuit design of the spike encoder.

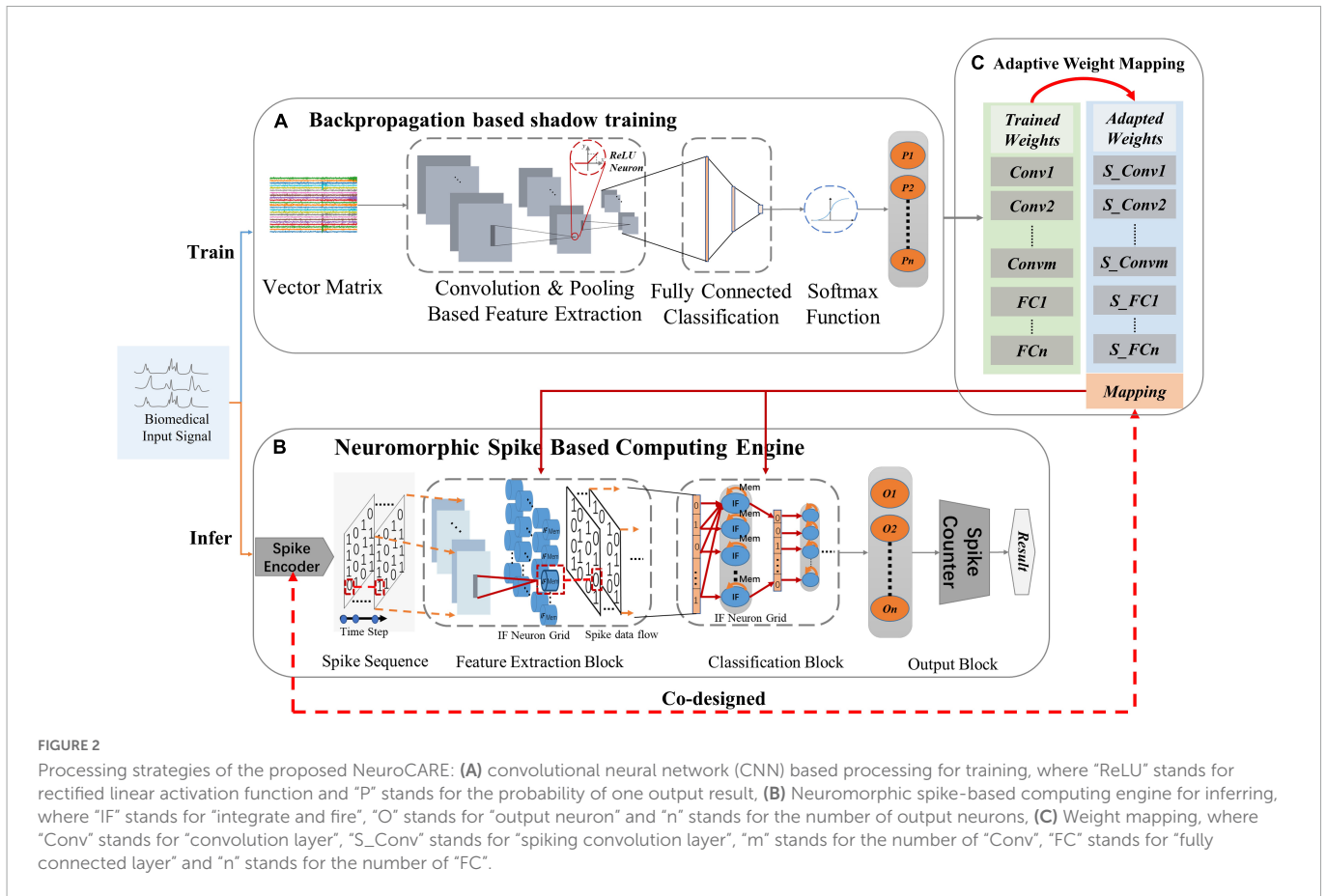
Algorithm 1 depicts the encoding process. The 2D input data sample is converted into a 3D vector with an extra time dimension. Meanwhile, key features of the original input are dispersed into multiple time steps after passing through the Gaussian spike encoder. The sparsity of the generated spike sequences is configurable utilizing several hyper-parameters: time-step, V_{th_up} and V_{th_down} . Time-step determines the length of time while V_{th_up} and V_{th_down} set a borderline for the generated Gaussian random values. The input original data for the spike encoder is raw biomedical sample. The encoded spikes are generated once at each time step *via* the proposed temporal Gaussian random discretization method. To determine elements of the encoded spikes, a Gaussian random matrix is generated at each time step. Holding the same shape as the original input sample, the generated matrix consists of random values following Gaussian distribution, whose variance is set as 1 and mean is set as $(V_{th_up} + V_{th_down})/2$. Elements of the generated random matrix is then one2one compared with the original input data. If the original element is greater, the corresponding encoded spike value is set one. Similarly, the spike value is set zero if the generated one is greater. This process applies to every time step, which means that the spike encoder generated one spike sample at each time step. If a vector in the shape of (C, H, W) (C, H, W stand for channel, height, and width respectively) is given as the input of the spike encoder, the output shall be (T, C, H, W) (T stands for time step) consisting of 0 or 1 elements.

```

1 : Input: time_step, original_data,  $V_{th\_up}$ ,
       $V_{th\_down}$ 
2 : Output: encoded spike sequences
3 : for  $t = 1 : time\_step$  do
4 :  $sample(t) \leftarrow$  Gaussian Random,
       $mean = \frac{V_{th\_up} + V_{th\_down}}{2}$ 
5 : end for
6 : for  $t = 1 : time\_step$  do
7 : for data_value in original data do
8 : if  $sample(t, data\_value) <$ 
       $original\_data(data\_value)$ 
9 :  $encoded\_spikes(t) \leftarrow 1$ 
10: else
11:  $encoded\_spikes(t) \leftarrow 0$ 
12: end if
13: end for

```

Algorithm 1 | Temporal Gaussian random sparse encoder.



The key benefits of spike encoder lie in that: (a) it discretized the information of raw biomedical data onto the time domain with adaptive sparsity, enabling the accurate and efficient Spiking-CNN processing method; (b) it helps to reduce the length of dataflow processed in the network from multi-bit (typically 8b/16b/32b) to merely 1-bit, and meanwhile still keeps the major features of input data so that the computation complexity and the required data memory is greatly reduced. The accumulation of the encoded spike sequences over all the time steps represents the retained information of input data. More data information is retained as the *time step* is set larger, which meanwhile brings more burden in memory and computing resources. In this work, different time steps are evaluated to compare the performance.

2.2. Channel-wise spatial-temporal network structure

For the processed raw biomedical signals, there always exist thousands or hundreds of elements in X axis (time axis) while only a few in Y axis (channel axis). Different channel records electrophysiological activities in different parts of human bodies. Thus, it may decrease the classification performance if the time (X) and channel (Y) axis are mixed up (Xu et al., 2020). Therefore, a channel-wise neural network structure is proposed in this work to avoid mixing up data features from different channels. The channel-wise neural network topology adopts single-dimension feature extraction kernels in convolution modules and

max-pooling modules, which also brings convenience to edge hardware implementation.

As shown in Figure 2A, the structure of the CNN-based processing topology consists of three parts. The input vector matrices first go through the feature extraction part (FEP) which is composed of multiple convolution (Conv) layers and max-pooling (MP) layers. Rectified linear units (ReLU) are also connected after every feature extraction layer. The data after the feature extraction part is then sent to the classification part (CP) which is composed of fully connected (FC) layers. The outputs of FC layers are then processed in the SoftMax function part. The final outputs stand in the format of probability ranging in 0~1 and the greatest one delivers the result of classification.

Figure 2B shows the structure of the proposed spike-based computing engine is shown in. Apart from the spike encoder and the spike counter, the proposed neuromorphic engine is mainly composed of a Spiking-CNN transformed from the CNN topology, which consists of a feature extraction block (FEB) and a classification block (CB). The FEB and CB have the same structure as FEP and CP of the CNN topology, respectively. The changes lie in that all the ReLU layers are replaced with IF neuron grids, which is explained detailly in Part C of Section “2 Module and architecture designs.” In Spiking-CNN, the data flow between two layers is transformed into the form of spikes (single-bit 0/1), which also transforms the “expensive” multiplication and accumulation (MAC) operations in CNN into the “cheap” adding-only (ADD) operations. Thus, Spiking-CNN requires far fewer computation resources than CNN, making it area-friendly and energy-efficient.

Table 1 shows the comparison of typical metrics between CNN, typical SNN, and this work. Since Spiking-CNN process data on time domain, the computing procedure can be executed both sequentially and parallelly, which depends on the architecture and resource constraints of the hardware processing element (PE), thus providing more flexibility for edge implementation.

2.3. Neuromorphic neuron model

In the proposed architecture, neuromorphic neuron grid is used as activation layer within each block (**Figure 2B**). As shown in **Figure 3A**, a biological neuron uses multiple dendrites to receive and send information. There exist one axon and many axon terminals at the end of the axon that connect with the dendrites of other neurons to transmit information. Neural models in NNs are designed in the purpose to mimic the function of biological neurons.

For common CNNs, the neuron model consists of inputs, outputs, and computation units (**Figure 3B**). The input connection is analogous to the dendrites of the neuron while the output is analogous to the axon. The computation units of CNN neurons complete the operations of multiplication and accumulation (MAC). However, such neurons are utilized only once in one single computation process.

For Spiking-CNNs, various biology-inspired neuron models have been explored (Roy et al., 2019). To mimic the behaviors of biological nerves, the neuromorphic neuron receives multiple spike sequences as inputs and may produce multiple output spikes as shown in **Figure 3C**. $V(t)$ and V_{reset} stand for the values of membrane potential and resting potential of such neuron, respectively. When spike inputs arrive, $V(t)$ increases, and later returns to V_{reset} after firing an output spike.

The leaky integrate and fire (LIF) neuron is a typical model to emulate biology neural behaviors. The behavior of LIF neurons is depicted in **Figure 3D**. However, the dynamic leaky process makes it hard to implement LIF neurons in high-speed digital circuits. In this study, the integrate and fire (IF) neuron, a simplified model with an adaptive threshold voltage V_{th} , is utilized to propose a hardware-friendly approach. For spike based IF neurons, the membrane potential $V_i(t)$ is updated with time as shown in Eq. (1)

$$V_i(t) = V_i(t - 1) + Input(t) - Leakage \quad (1)$$

In IF neurons, $\sum \delta_j(t - t_j)$ describes the sum of pre-synaptic stimulus of each neuron which also stands as the accumulation of spike inputs (**Figure 3C**) and leakage of IF neurons is set to be a constant voltage ΔV as show in Eq. (2)

$$V_i(t) = V_i(t - 1) + \sum_j w_{ij} \sum_{t_j \in T} \delta_j(t - t_j) - \Delta V \quad (2)$$

As shown in **Figure 3E**, when the neuron membrane potential $V_i(t)$ exceeds the threshold voltage V_{th} , a spike is fired at the output. Then $V_i(t)$ returns to the resting potential V_{reset} .

2.4. Processing dataflow

To process raw biomedical data in the proposed neuromorphic engine, the original input data is first encoded into time-dependent spike sequences as described in Part A of Section “2

Module and architecture designs.” *Time Step* is first chosen. Other hyperparameters of spike encoder— V_{th_up} and V_{th_down} are set and adjusted along with the original input data distributions to achieve an optimal encoding performance.

At each point of time step, the encoded inputs are processed by the Spiking-CNN, where some of the output neurons may be activated and fire output spikes. The numbers of output spikes are recorded by the spike counter over all the time steps (**Figure 2B**) and the greatest one produces the result of classification.

In this study, the V_{reset} of all the IF neurons are set as 0 for the convenience to implement the whole neuromorphic engine on edge hardware. Threshold voltages V_{th} are set the same for each layer and adjusted along with the V_{th_up} and V_{th_down} to improve the performance.

2.5. Adaptive weight mapping configuration

As for the weight mapping which is shown in **Figure 2C**, the trained CNN weights are mapped on the Spiking-CNN in a way2way fashion after scaling with an adaptive scaling factor f .

Based on statical data distribution of activations in the proposed neural networks, we propose an adaptive weight mapping configuration method to keep high-performance classification for the conversion from CNN to Spiking-CNN. In the proposed mapping configuration, the scaling factor f is adaptive with the hyper parameters determined for the Gaussian spike encoder, i.e., $time_step, V_{th_up}, V_{th_down}$.

For the L th-layer neural network, the weight vector connecting between the k_{th} neuron of layer $l - 1$ and the i_{th} neuron of layer l can be expressed as $W_{ik}^l, l \in \{1, \dots, L\}$. For CNNs using ReLU neurons, the gained activation of neuron i in l layer is

$$a_i^l = Max\left(0, \sum_k^{K^{l-1}} W_{ik}^l * a_k^{l-1}\right) \quad (3)$$

where K^{l-1} represents the number of neurons in the $l - 1$ layer that are connected to the neuron i of l layer. Thus a_k^0 represents the k th value of the input data. Following the Gaussian distribution, the probability that the encoded spike value is 0 at the time step t is

$$P(S_{kt}^0 = 0) = \int_{V_{th_down}}^{a_k^0} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-mean)^2}{2}\right) dx \quad (4)$$

Similarly, the probability that the encoded spike value is 1 at the time step t is

$$P(S_{kt}^0 = 1) = \int_{a_k^0}^{V_{th_up}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-mean)^2}{2}\right) dx \quad (5)$$

where

$$mean = \frac{V_{th_up} + V_{th_down}}{2} \quad (6)$$

Therefore, the accumulated value of the encoded data a_k^0 through all the time steps is

$$\begin{aligned} Vac_k^0 &= \sum_t^T 0 * P(S_{kt}^0 = 0) + 1 * P(S_{kt}^0 = 1) \\ &= T \int_{a_k^0}^{V_{th_up}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-mean)^2}{2}\right) dx \end{aligned} \quad (7)$$

TABLE 1 Overall comparison between convolutional neural network (CNN), spiking neural network (SNN), and this work.

	Comparison metrics	CNN	SNN	This work
Network structure	No. of FE Layer	M_c	M_s	$M'_s (= M_c)$
	Function layer	Softmax, ReLU	Delta/rate encoder, spike counter	Sparse Gaussian encoder, spike counter
	Input	2-D vector	Spike sequence (long time steps > 100)	Spike sequence (few time steps ~10/25)
	Output	Probability (multi-bit FP)	Number of spikes (INT)	Number of spikes (INT)
Neuron model	Activation function	ReLU function	LIF activation	IF activation
	Weight	w_{ij}	w_{ij}	w'_{ij} (Adaptive mapping)
	Bias	Vary between deff. layers	Vary between deff. layers	Constant ΔV
	Spike-based	-	✓	✓
	Dynamic leakage	Not required	Required	Not required
	FE domain	Spatial	Temporal	Spatial-temporal
Operation		Multiplication and accumulation (MAC)	Accumulation (ADD), dynamic leakage	Accumulation only (ADD)
Dataflow		Multi-bit value	Single-bit 0 or 1	Single-bit 0 or 1
Weight resolution		Multi-bit value		

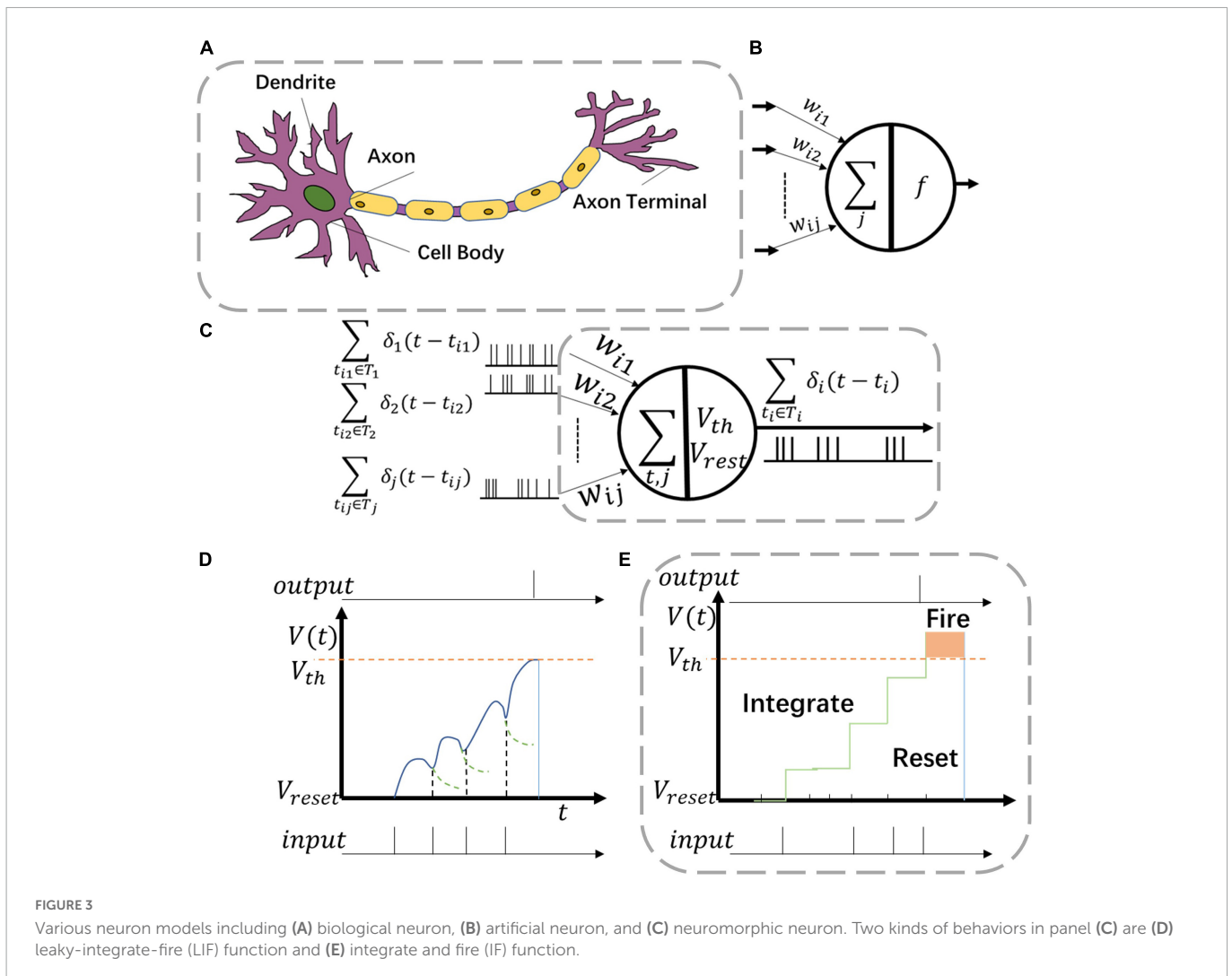


FIGURE 3

Various neuron models including (A) biological neuron, (B) artificial neuron, and (C) neuromorphic neuron. Two kinds of behaviors in panel (C) are (D) leaky-integrate-and-fire (LIF) function and (E) integrate and fire (IF) function.

Then we can get the scaling factor of the first layer

$$f_1 = \frac{1}{K^0} \sum_k \frac{a_k^0}{Vac_k^0} = \frac{1}{K^0} \sum_k \frac{a_k^0}{T \int_{a_k^0}^{V_{th_up}} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{(x-mean)^2}{2}\right) dx} \quad (8)$$

For the following layers of Spiking-CNN, considering the integrate-and-fire function of IF neurons, the accumulated value of the neuron i of l layer through all the time steps is

$$Vac_i^l = \sum_t \left(\sum_k^{K^{l-1}} f_i^l W_{ik}^l * S_{kt}^{l-1} - r^l * V_{th} + V_{reset} \right) \quad (9)$$

$$\cong T * \left(\sum_k^{K^{l-1}} f_i^l W_{ik}^l * r^l - r^l * V_{th} + V_{reset} \right)$$

where f_i^l stands for the scaling factor for neuron i in l layer and r^l stands for the firing rate of the l layer which is defined as

$$r^l = 0.5 * \left(1 - \frac{l}{L} \right) \quad (10)$$

Combining Eq. (3), Eq. (9), and Eq. (10), we can get the scaling factor of the l -th layer in Eq. (11)

$$f_l = \frac{1}{K^{l-1}} \sum_i \frac{\frac{1}{T} * \sum_k^{K^{l-1}} W_{ik}^l * a_k^{l-1} + 0.5 * \left(1 - \frac{l}{L} \right) * V_{th} - V_{reset}}{\sum_k^{K^{l-1}} W_{ik}^l * a_k^{l-1}} \quad (11)$$

3. Experiments and evaluation results

Evaluations of NeuroCARE are conducted in terms of classification performance, energy consumption, and resource expenses in this study. We deploy this framework on a simulator based on Eyeriss architecture (Chen et al., 2017) and conduct three edge healthcare applications to demonstrate its advantages and versatility: (1) epileptic seizure prediction based on EEG, (2) arrhythmia detection based on ECG, and (3) hand gesture classification based on EMG. Therefore, three different benchmark datasets are utilized to evaluate the performance of the proposed approach.

3.1. EEG based seizure prediction

The CHB-MIT EEG dataset is used to evaluate the performance of NeuroCARE on EEG-based seizure prediction tasks. The CHB-MIT dataset contains 23 measurements of scalp EEG data recorded from 22 patients at 256 Hz sampling rate (Shoeb, 2009). Among the total 23 measurements, 15 are recorded under the same fixed signal acquisition configuration using 23 electrodes, while the remaining 8 have some changes.

For seizure prediction tasks based on EEG signals, a tiny time window between the end of preictal interval and the beginning of seizure onset is defined as the seizure prediction horizon (SPH). The intervals chosen for the experiments are determined by SPH and the preictal interval length (PIL). In this experiment, we choose to use 30 min PIL and 5 min SPH. Using a fixed time window of 20 s, two categorized samples of the interictal intervals and the preictal intervals are extracted from the CHB-MIT dataset. For every extracted sample, the data height (H) is the number of recording channels, and the data width (W) is sampling-rate * 20 s, where each

element is the recorded voltage value. Thus, the input vector shape (H, W) is 23 * 5,120 for this experiment. However, the interictal intervals contained in CHB-MIT dataset are much more than preictal intervals causing the problem of sample imbalance in training, which may lead to poor performances (Japkowicz and Stephen, 2002; Barandela et al., 2004). To overcome this barrier, samples of preictal intervals are extracted using 5 s overlapping (Tian et al., 2021).

In this study, only lead seizures occurring at least 4 h after the previous ones are taken into consideration (Hussein et al., 2019). Therefore, there are totally seven subjects suitable for the experiment. In this work, the results of evaluated metrics are the mean values of all the seven subjects.

With the ratio of 4:1, we separate the EEG dataset into training dataset and testing dataset randomly. The training dataset is used to gain the trained weights in CNN-based topology. The trained weights are first restored, then scaled, and finally mapped on the spike-based computing engine correspondingly. The performance of NeuroCARE is recorded on the testing dataset. When operating predictions, the raw EEG data is first transformed into spike sequences via the Gaussian spike encoder. The Spiking-CNN mapped with the scaled weights processes the encoded spike sequences and then produces results of prediction. Thus, the performance can be evaluated by comparing the results and the labels.

The neural network structure for EEG processing is detailed in Figure 4A. The proposed network is composed of five spiking convolution and max-pooling (SC and MP) layers, two spiking fully connected (SFC) layers, and two extra layers of spike encoder and spike counter.

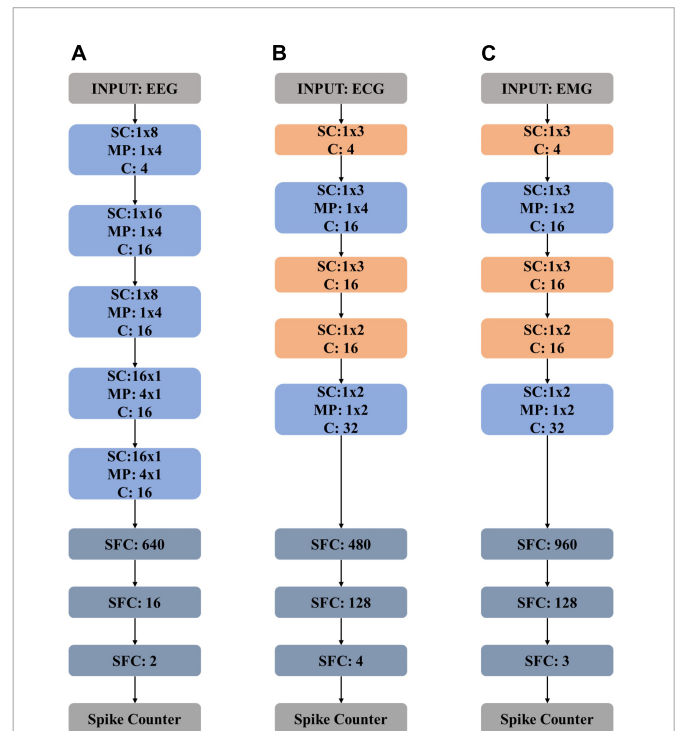


FIGURE 4 Model structures of (A) electroencephalography (EEG) network, (B) electrocardiography (ECG) network, and (C) electromyography (EMG) network. “SC” stands for “spiking convolution”, “MP” stands for “max pooling”, “SFC” stands for “spiking fully connected” and “C” stands for “channel”.

3.2. ECG based arrhythmia detection

Evaluations on the performance of ECG-based arrhythmia detection tasks are conducted using MIT-BIH dataset in this work, which are obtained from total 47 subjects studied by the MIT-BIH Arrhythmia Laboratory. The MIT-BIH dataset contains forty-eight 30-min excerpts of two-channel ambulatory ECG recordings. Among the total 48 recordings, 23 are chosen randomly from 4k 24-h ambulatory ECG samples. These samples were recorded from a mixed collection of inpatients and outpatients with the ration of about 3:2 at Boston’s Beth Israel Hospital. The left 25 recordings include less common but clinically significant arrhythmia samples that cannot be well-represented in randomly chosen datasets, which are selected from the same set (Goldberger et al., 2000).

All the recordings are digitally sampled with a 11-bit resolution over a 10-mV range at 360 Hz per channel. As defined in standards from Association for the Advancement of Medical Instrumentation (AAMI), ECG recordings can be divided into five classes: ventricular ectopic beats (VEB), supraventricular ectopic beats (SVEB), fusion beats (F), non-ectopic beats (N), and unknown beats (Q). In total, MIT-BIH dataset contains 90,081 N samples, 7,008 VEB samples, 2,781 SVEB samples, 802 F samples, and 15 Q samples. In this experiment, to overcome the problem of data unbalance, we randomly selected 800 samples from N, VEB, SVEB, and F, respectively. Q samples are excluded in this experiment for there are too few to use for training. The total 3,200 samples are further separated into training dataset and testing dataset randomly in a 4:1 ratio.

The proposed ECG network structure is detailed in Figure 4B, which is composed of five spiking convolution (SC) layers, two max-pooling (MP) layers, and two spiking full-connected (SFC) layers. A spike encoder layer and a spike counter layer are also needed. The shadow weights of the CNN-based topology are first trained on the training data and then scaled and mapped on the spike-based computing engine correspondingly. The classification performance of the spike-based ECG network is then tested. The method of data processing in the inference procedure remains the same as the way mentioned in Part A of Section “3 Experiments and evaluation results.”

3.3. EMG based hand gesture recognition

To evaluate the performance on EMG based hand gesture recognition task, Nina Pro DB1 benchmark is used in this experiment. Nina Pro DB1 contains surface EMG (sEMG) data collection of 10 electrodes of measurements from 27 subjects (Atzori et al., 2014). All the data is collected with a sampling rate of 100 Hz. When collecting data, each type of hand gesture is repeated 10 times in a row. Every movement of hand gesture lasts for 5 s, and there exists a 3 s resting time between every two movements. Recordings of three kinds of commonly used hand gestures are utilized for evaluation of performance in this experiment, including the resting position, the thumb up position and the index flexion position. With the ratio of 4:1, the recorded samples are randomly separated into training dataset and testing dataset, respectively.

Input data is first encoded into spike sequences, and then processed by the proposed SNN. The size of the input sEMG samples during training and inference is 12 × 10 with a batch size of 32. Before

converting into spike sequences in the time domain, the original samples are also min-max normalized into the range of (0,1).

Figure 4C shows the network structure. The proposed neural network consists of five spiking convolution (SC) layers, two max-pooling (MP) layers, and two spiking full-connected (SFC) layers with a spike encoder and a spike counter. In the training procedure, the training data is used to train the proposed CNN-based topology to get weights which are then scaled and mapped on the spike-based computing engine for inferring. The testing data is used to evaluate the performance of the spike-based engine.

3.4. Performance evaluations

In this study, to evaluate the classification performance of the proposed neuromorphic approach, we choose the standard evaluation metrics including accuracy, precision, recall, sensitivity, F1 score, and an overall score for the above-described multi-class (N classes) classification tasks, which are defined as follows:

$$Accuracy (Acc) = \frac{\sum_{i=0}^{N-1} Class_{i,correct}}{\sum_{i=0}^{N-1} Class_{i,total}} \tag{12}$$

$$Precision = \frac{1}{N-1} \sum_{i=1}^{N-1} \frac{Class_{i,correct}}{Class_{i,correct} + Class_{0,false}} \tag{13}$$

$$Recall = \frac{1}{N-1} \sum_{i=1}^{N-1} \frac{Class_{i,correct}}{Class_{i,correct} + Class_{0,correct}} \tag{14}$$

$$Sensitivity (Sen) = \frac{Class_{0,correct}}{Class_{0,total}} \tag{15}$$

$$F1\ Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{16}$$

$$Overall\ Score = \frac{Acc + Sen + F1\ Score}{3} \tag{17}$$

where $Class_{0,total}$ stands for the number of the normal state or the resting state and $Class_{i,total}$ stands for the number of the i_{th} class of the processed biomedical data.

The effect of the set number of *time step* (TS) of the proposed spike encoder is shown in Figure 5, where the larger TS is, the higher accuracy is gained.

For the evaluation of computation efficiency of the proposed approach, the number of computing operations (ADD and MUL) and the required weight memory are estimated. The metric of *computation complexity* (TC) is also calculated to quantize the reduction of computation costs brought by the proposed approach. The calculation of TC of CNN and the proposed Spiking-CNN are presented in Eq. (18) and Eq. (19), respectively

$$TC_{CNN} = \sum_{all\ layers} M_H M_W (K_H K_W + K_H + K_W - 1) C_{in} C_{out} \times Ops \times bit + \sum_{all\ layers} N_{in} N_{out} \times Ops \times bit \tag{18}$$

$$TC_{SCNN} = \sum_{all\ layers} M_H M_W (K_H + K_W - 1) C_{in} C_{out} \times Ops \times bit \times TS + \sum_{all\ layers} N_{in} N_{out} \times Ops \times bit \times TS \tag{19}$$

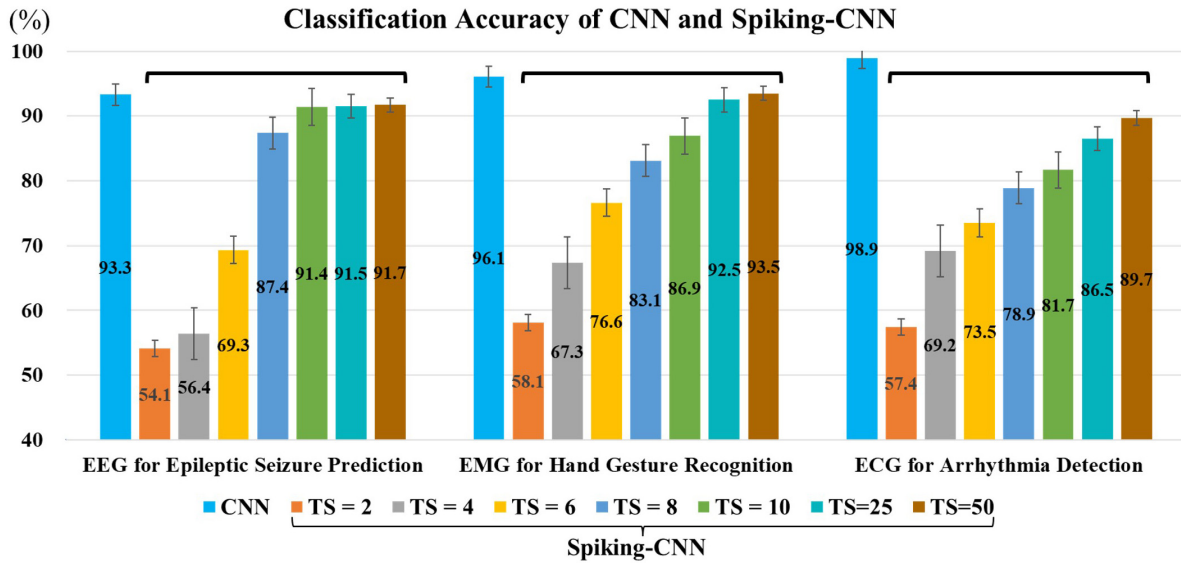


FIGURE 5 Classification accuracies for electroencephalography (EEG), electromyography (EMG), and electrocardiography (ECG) utilizing convolutional neural network (CNN) and Spiking-CNN with different configurations in time step, where “TS” stands for “the number of time steps” for Spiking-CNN. The larger TS is set, the higher accuracies are gained.

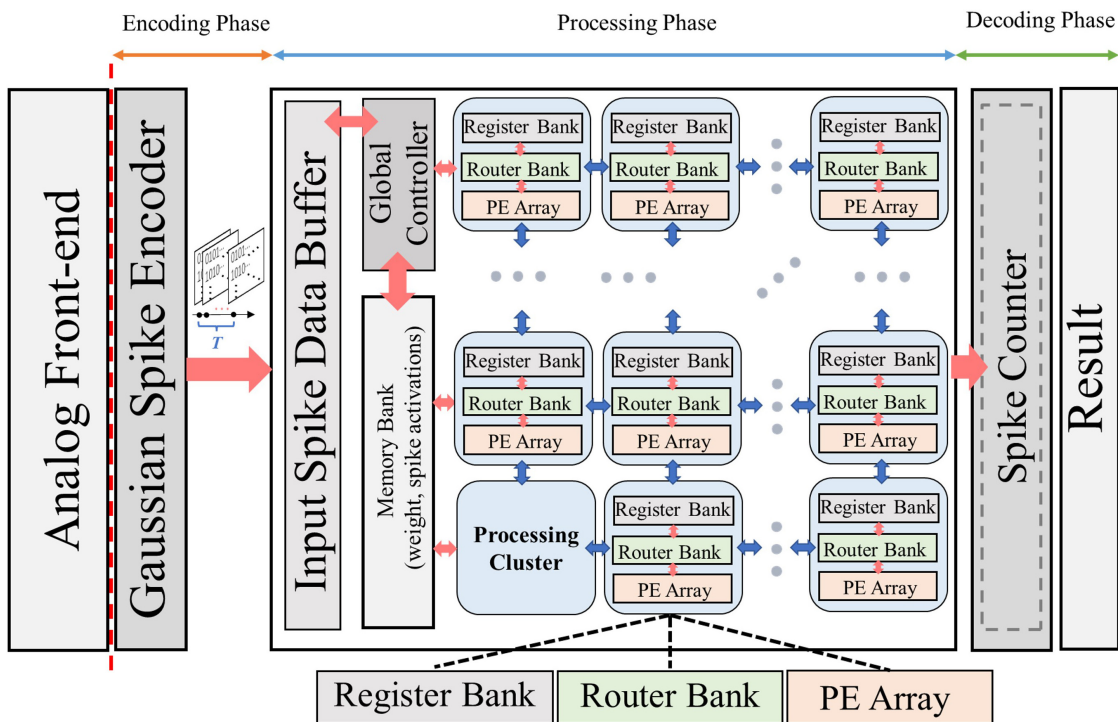


FIGURE 6 Overall system architecture design of NeuroCARE, consisting of encoding phase, processing phase and decoding phase.

where M , N , K , W , H , and C stand for *Feature Map Size in Conv Layers*, *Neuron Number in FC Layers*, *Kernel*, *Width*, *Height*, and *Number of Channels*; Ops is the number of required cycles to finish one computing operation [conditional branching: 1, ADD: 1, MUL: 3, and MAC: 4 based on edge deployment (Hennessy and Patterson, 2011)]; *bit* stands for needed bits of activation buffer between two NN layers in hardware. In SNN,

the required operations are ADD and conditional branch (used to judge if the output spike should be fired). In the contrast, CNN needs more complicated MAC (MUL plus ADD operations). SNN processes single-bit activations while multi-bit (8b/16b/32b) activations are typically used in CNN implementations. To make a conservative evaluation of our design, we choose 8b for CNN in the calculation.

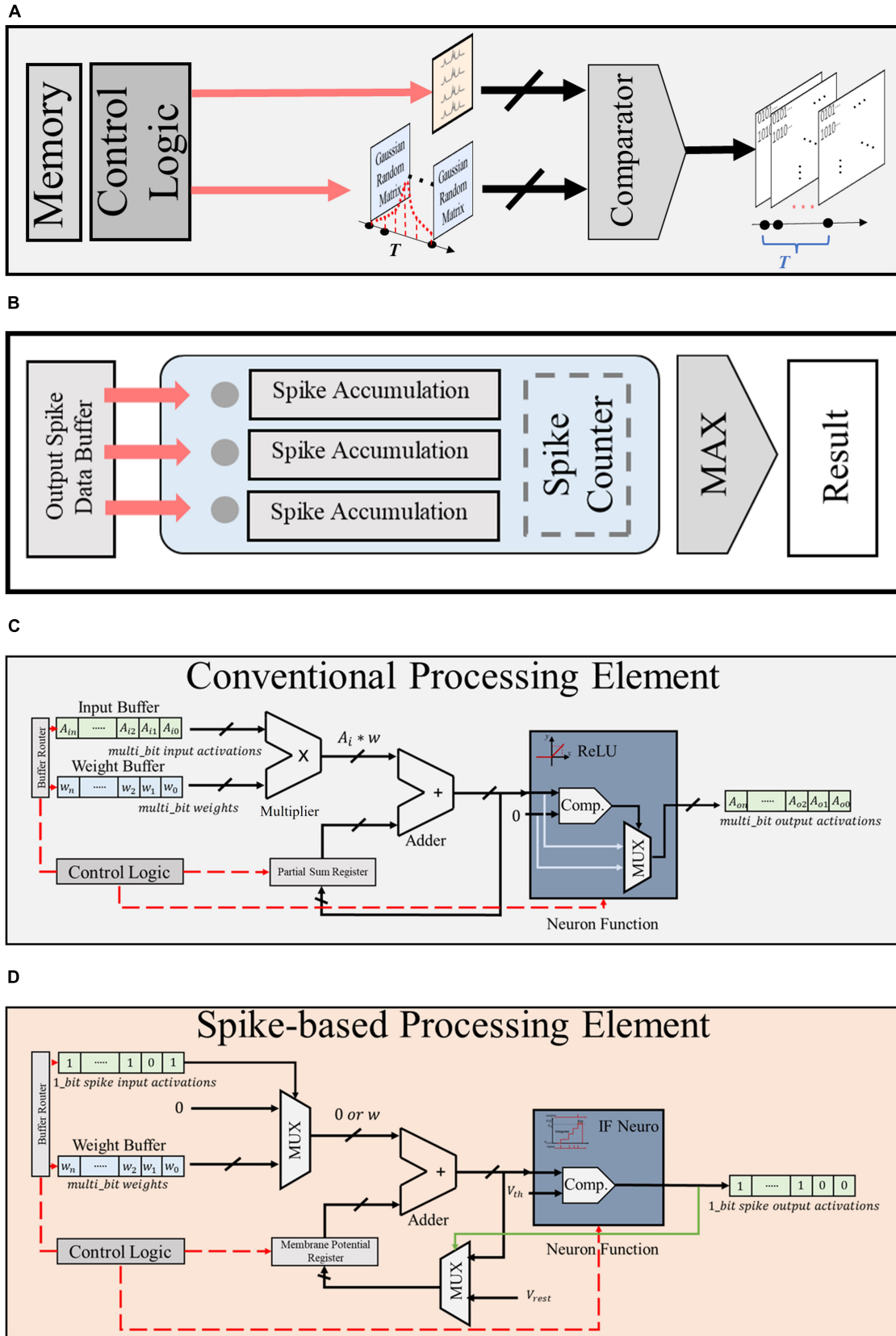


FIGURE 7 Module design of (A) Gaussian spike encoder, (B) output spike counter, (C) conventional CNN-based PE, and (D) proposed spike-based PE.

In the EEG experiment, our approach achieves an accuracy of 91.5% with a 96.1% reduction in computation complexity when the time step of spike-encoder is set 25. In the ECG experiment, with the same setting of time step, the proposed approach achieves an accuracy of 86.5% with an 80.5% reduction in computation complexity. In the EMG experiment, with the time step of 25, the proposed Spiking-CNN achieves an accuracy of 92.5% with an 80.7% reduction in computation complexity.

3.5. System implementation

Figure 6 shows the baseline architecture used to implement the proposed framework. Various kinds of bio-signals are first encoded into spike sequences by the Gaussian spike encoder. The encoded spike data are then sent to the processing hardware and stored in the input spike data buffer. The architecture proposed in Eyeriss (Chen et al., 2017) is adopted in our work. It is optimized for CNN topology, hence suitable for the implementation of NeuroCARE. The output spike activations from the processing hardware are then sent to the spike counter to produce the result of classification. The proposed algorithm can be processed in the same way as demonstrated in Eyeriss (Chen et al., 2017). A global controller is used to control the data fetching and processing. All the processing is done in the processing cluster (PCL) arrays. Each PCL consists of a register bank, a router bank, and a processing element (PE) array. A memory bank is also connected with each PCL, which stores weights of the neural network and the spike activations between different network layers. Detailed designs of PEs are shown in Figure 7. Compared to other state-of-art works, our proposed method has two main advantages: (1) No multiplexer is required in the PCL of NeuroCARE, which greatly reduces the energy consumption in processing phase; (2) The bit resolution of activation data buffers in NeuroCARE is single-bit while conventional CNN-based implementations use multi-bit, thus our design requires much less energy consumption in data access.

3.6. Efficiency evaluations

To evaluate the implementation and energy efficiency of the proposed approach, key modules in NeuroCARE are designed and verified in the register-transfer level (RTL) level.

As presented in Figure 7A, the proposed Gaussian spike encoder is composed of a data buffer that stores the random Gaussian matrixes, a comparator and a control module. The controller fetches data from the buffer while the comparator compares the Gaussian value with the raw bio-signal to generate spike sequences. The proposed Gaussian spike encoder is practical with a very simple structure.

The design of spike counter is described in Figure 7B, which also has a simple structure. As mentioned in Part E of Section “3 Experiments and evaluation results,” the output spike activations are sent to the spike counter and stored in the output spike data buffer. When the whole processing procedures are completed through all the time steps, the total firing times of all the output neurons are calculated by the spike counter and the greatest one produces the result of classification.

Figures 7C, D present the detailed design of conventional CNN-based PE (CPE) and the proposed spike-based PE (SPE), respectively. As presented, the SPE has 1 adder, 2 mux, and 1 comparator, while the CPE requires 1 adder, 1 mux, 1 comparator, and 1 multiplexer. Compared to CPE, SPE shows advantages in two aspects: (1) the resolution of input and output buffers in SPE is single bit, thus brining reduction in memory assess and energy consumption; (2) the structure of SPE is much simpler and requires no multiplexer, thus further reduce the hardware cost and resource budget.

Evaluations on the energy consumption and area occupation of NeuroCARE system are conducted based on statistics from Horowitz (2014). The technology is assumed to be 45 nm and the voltage supply is assumed as 0.9 V. Thus, we get the comparison of energy consumption and area occupation between SPE and CPE under different data resolutions, which is shown in Table 2. The results demonstrate that using the same configuration, spike-based processing could achieve over 80% reduction in energy consumption and over 64.8% reduction in area occupation.

3.7. Overall evaluation

To evaluate the overall performance with considerations of both classification performance and resource expenses, we define a figure-of-merit (FOM) in this work as given in Eq. (20)

$$FOM = \frac{\text{Overall Score} \cdot \text{EA factor}}{(\text{MUL} + \text{ADD} + \text{MEM})} \quad (20)$$

TABLE 2 Comparison of energy and area consumption between CNN-based PE (CPE) and spike-based PE (SPE).

Metrics	CPE	SPE
Technology		45 nm
Voltage supply		0.9 V
Weight data width	8b int / 32b int / 16b fp / 32b fp	8b int / 32b int / 16b fp / 32b fp
Activation data width	8b int / 32b int / 16b fp / 32b fp	1 bit
Energy per op.	0.25 pJ / 3.3 pJ / 2.0 pJ / 5.5 pJ	0.03 pJ / 0.1 pJ / 0.4 pJ / 0.9 pJ
Area	349 μm^2 / 3,632 μm^2 / 5,824 μm^2 / 11,884 μm^2	36 μm^2 / 137 μm^2 / 1,360 μm^2 / 4,184 μm^2
Energy reduction ration		88% / 96.7% / 80% / 83.6%
Area reduction ratio		89% / 96.2% / 77% / 64.8%
Efficiency acceleration		7.58\times / 79.7\times / 2.17\times / 1.79\times

Energy reduction ration (ER) = 1 - SPE energy / CPE energy. Area reduction ration (AR) = 1 - SPE area / CPE area. Efficiency acceleration (EA) = 1 / [time step * (1 - ER) * (1 - AR)]. Bold values represent the advantages of the proposed method.

TABLE 3 Comparison of performance with other works.

Classification task type	References	Model type	Dataset	No. of classes	Data type	Network depth	Channel wise	Overall score (%)	Accuracy (%)	F1 score	Number of ADD	Number of MUL	Memory occupied	FOM (%/M)
EEG based epileptic seizure prediction	NN2018 (Truong et al., 2018a)	STFT-CNN	MIT-CHB	2 class	Pre-processed	9 layers	No	91.7	81.2	81.4	1.17M	1.45M	3.81M	11.6
	BIBM2018 (Eberlein et al., 2018)	Wavelet-CNN	iEEG data	2 class	Pre-processed	21 layers	No	74.5	77.0	73.7	1.74M	2.20M	1.70M	16.26
	JBHI2020 (Zhang et al., 2020)	CSP-CNN	MIT-CHB	2 class	Pre-processed	7 layers	No	90.0	90.0	91.0	1.43M	1.97M	2.27M	15.9
	ISCAS2020 (Zhao et al., 2020)	BSD-CNN	MIT-CHB	2 class	Raw data	13 layers	Yes	94.33	97.0	94.69	5.54M	0.61M	0.067M	15.18
	This work	CNN	MIT-CHB	2 class	Raw data	13 layers	Yes	95.2	93.3	97.0	2.39M	2.84M	0.33M	17.12
		Spiking-CNN*						92.7	91.4	95.1	2.39M	0		61.0
ECG based arrhythmia detection	TBME2016 (Kiranyaz et al., 2016)	CNN	MIT-BIH (20 records)	5 class	Raw data	5 layers	No	87.6	98.3	76.4	0.25M	0.34M	0.64M	71.2
	ICHI2018 (Kachuee et al., 2018)	Residual CNN	PTB	5 class	Pre-processed	19 layers	No	95.4	95.9	95.1	4.83M	5.03M	1.33M	8.5
	BDAI2018 (Zhang et al., 2018)	LSTM	MIT-BIH	5 class	Pre-processed	12 layers	No	97.6	97.7	97.6	3.05M	2.71M	1.29M	13.8
	This work	CNN	MIT-BIH	4 class	Raw Data	10 layers	Yes	98.4	98.9	97.4	0.42M	0.61M	0.26M	77.1
			Spiking-CNN*						96.7	86.5	98.2	0.42M	0	
EMG based Hand Gesture Recognition	IJCAI2017 (Du et al., 2017)	3D-CNN	Nina Pro	52 class	Pre-processed	8 layers	No	79.5	79.4	79.6	0.528M	0.530M	4.8M	13.6
	Sci. Rep.2016 (Geng et al., 2016)	Filter + CNN	Nina Pro	52 class	Pre-processed	9 layers	No	92.5	96.7	88.9	0.522M	0.524M	2.4M	18.9
	Frontiers2016 (Atzori et al., 2016)	Filter + CNN	Nina Pro	52 class	Pre-processed	7 layers	No	66.6	75.32	58.0	4.3M	4.3M	0.091	7.7
	This work	CNN	Nina Pro	3 class	Raw data	10 layers	Yes	95.5	96.1	94.9	0.57M	0.78M	0.51M	51.3
			Spiking-CNN*						85.7	92.5	82.7	0.57M	0	

*The TS for spiking-CNN is set as 25 for comparison.

According to **Table 2**, the efficiency acceleration (EA) factor ranges in 1.79 ~ 79.7. To make a conservative evaluation of our design, we choose this scaling factor as 1.79.

The comparison between achieved results in this work and prior-art publications on metrics mentioned above is shown in **Table 3**. As reported, the proposed NeuroCARE shows advantages in both overall performance and energy efficiency compared to reported works. This approach stands as a channel-wise method without utilizing any extra feature extraction techniques, achieving high overall scores evaluated on three benchmarks. It also requires no MUL operations and little occupied memory, which makes it energy efficient. Therefore, NeuroCARE manages to achieve the highest FOM compared among all the other works.

4. Conclusion

In this study, we propose NeuroCARE, a generic edge neuromorphic framework for healthcare applications. Modules include Gaussian spike encoder, neuromorphic neuron model, processing strategies, and the adaptive weight mapping method. To achieve high performance and reduce the required hardware resources, we first implemented CNN topologies, then scale and map the trained weights on Spiking-CNN correspondingly. Raw biomedical signals are processed in neuromorphic fashion on the time domain which significantly reduces the required memory and computation complexity.

As demonstrated by the implementation results, NeuroCARE achieves state-of-the-art classification results in epileptic seizure prediction, arrhythmia detection, and hand gesture recognition tasks, while managing to reduce the computation complexity by over 80.7%, reduce the computation energy consumption by over 80%, and reduce the area occupation by over 64.8% compared to CNN based methods, which greatly empowers the ultra-efficient neuromorphic intelligence for edge biomedical applications.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found here: The EEG dataset for this study can be found

in the CHB-MIT Dataset <https://physionet.org/content/chbmit/1.0.0/>. The ECG dataset for this study can be found in the MIT-BIH Arrhythmia Database <https://physionet.org/content/mitdb/1.0.0/>. The EMG dataset for this study can be found in the Ninapro DB1 Dataset <http://ninaweb.hevs.ch/node/3>.

Author contributions

FT and JY designed the architecture and key modules of NeuroCARE. FT conducted all the relevant experiments and implementations. SZ preprocessed the dataset. MS supervised the study. All authors discussed the results, wrote the manuscript, and approved the submitted version.

Funding

This work was supported by Westlake University and the Zhejiang Key R&D Program Project No. 2021C03002.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

- Abiri, R., Borhani, S., Sellers, E. W., Jiang, Y., and Zhao, X. (2019). A comprehensive review of EEG-based brain-computer interface paradigms. *J. Neural Eng.* 16:011001.
- Amirshahi, A., and Hashemi, M. (2019). ECG classification algorithm based on STDP and R-STDP neural networks for real-time monitoring on ultra low-power personal wearable devices. *IEEE Trans. Biomed. Circuits Syst.* 13, 1483–1493. doi: 10.1109/TBCAS.2019.2948920
- Atzori, M., Cognolato, M., and Müller, H. (2016). Deep learning with convolutional neural networks applied to electromyography data: A resource for the classification of movements for prosthetic hands. *Front. Neurobot.* 10:9. doi: 10.3389/fnbot.2016.00009
- Atzori, M., Gijssberts, A., Castellini, C., Caputo, B., Hager, A.-G. M., Elsig, S., et al. (2014). Electromyography data for non-invasive naturally-controlled robotic hand prostheses. *Sci. Data* 1:140053.
- Barandela, R., Valdovinos, R. M., Sánchez, J. S., and Ferri, F. J. (2004). "The imbalanced training sample problem: Under or over sampling?," in *Joint IAPR international workshops on statistical techniques in pattern recognition (SPR) and structural and syntactic pattern recognition (SSPR)*, eds A. Fred, T. M. Caelli, R. P. W. Duin, A. C. Campilho, and D. de Ridder (Berlin: Springer), 806–814.
- Cao, Y., Yang, C., and Deepak, K. (2015). Spiking deep convolutional neural networks for energy-efficient object recognition. *Int. J. Comput. Vis.* 113, 54–66.
- Chen, Y., Dai, C., and Chen, W. (2019). "A real-time EMG-controlled functional electrical stimulation system for mirror therapy," in *Proceedings of the 2019 IEEE biomedical circuits and systems conference (BioCAS)*, (Piscataway, NJ: IEEE), 1–4. doi: 10.1109/BIOCAS.2019.8919069
- Chen, Y., Krishna, T., Emer, J. S., and Sze, V. (2017). Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks. *IEEE J. Solid State Circuits* 52, 127–138. doi: 10.1109/JSSC.2016.2616357
- Chen, Y.-H., and Sawan, M. (2021). Trends and challenges of wearable multimodal technologies for stroke risk prediction. *Sensors* 21:460. doi: 10.3390/s21020460
- Davidson, S., and Furber, S. (2021). Comparison of artificial and spiking neural networks on digital hardware. *Front. Neurosci.* 15:651141. doi: 10.3389/fnins.2021.651141

- Diehl, P. U., Neil, D., Binas, J., Cook, M., Liu, S.-C., and Pfeiffer, M. (2015). "Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing," in *Proceedings of the 2015 international joint conference on neural networks (IJCNN)*, (Piscataway, NJ: IEEE), 2015.
- Donati, E., Payvand, M., Risi, N., Krause, R., and Indiveri, G. (2019). Discrimination of EMG signals using a neuromorphic implementation of a spiking neural network. *IEEE Trans. Biomed. Circuits Syst.* 13, 795–803.
- Du, Y., Wong, Y., Jin, W., Wei, W., Hu, Y., Kankanhalli, M. S., et al. (2017). "Semi-supervised learning for surface EMG-based gesture recognition," in *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17)*, Melbourne, VIC.
- Eberlein, M., Hildebrand, R., Tetzlaff, R., Hoffmann, N., Kuhlmann, L., Brinkmann, B., et al. (2018). "Convolutional neural networks for epileptic seizure prediction," in *Proceedings of the 2018 IEEE international conference on bioinformatics and biomedicine (BIBM)*, (Piscataway, NJ: IEEE), 2577–2582.
- Geng, W., Du, Y., Jin, W., Wei, W., Hu, Y., and Li, J. (2016). Gesture recognition by instantaneous surface EMG images. *Sci. Rep.* 6:36571.
- Ghosh-dastidar, S., and Adeli, H. (2009). Spiking neural networks. *Int. J. Neural Syst.* 19, 295–308.
- Goldberger, A. L., Amaral, L. A., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., et al. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation* 101, e215–e220.
- Guo, L., Wang, Z., Cabrerizo, M., and Adjouadi, M. (2017). A cross-correlated delay shift supervised learning method for spiking neurons with application to interictal spike detection in epilepsy. *Int. J. Neural Syst.* 27:1750002.
- Hennessy, J. L., and Patterson, D. A. (2011). *Computer architecture: A quantitative approach*. Amsterdam: Elsevier.
- Horowitz, M. (2014). "1.1 Computing's energy problem (and what we can do about it)," in *Proceedings of the 2014 IEEE international solid-state circuits conference digest of technical papers (ISSCC)*, (Piscataway, NJ: IEEE), 10–14. doi: 10.1109/ISSCC.2014.6757323
- Hu, Y., Wong, Y., Wei, W., Du, Y., Kankanhalli, M., and Geng, W. (2018). A novel attention-based hybrid CNN-RNN architecture for sEMG-based gesture recognition. *PLoS One* 13:e0206049. doi: 10.1371/journal.pone.0206049
- Hussein, R., Ahmed, M. O., Ward, R., Wang, Z. J., Kuhlmann, L., and Guo, Y. (2019). Human intracranial eeg quantitative analysis and automatic feature learning for epileptic seizure prediction. *arXiv [preprint]* arXiv:1904.03603
- Iyer, L. R., and Chua, Y. (2020). "Classifying neuromorphic datasets with tempotron and spike timing dependent plasticity," in *Proceedings of the 2020 international joint conference on neural networks (IJCNN)*, (Glasgow), 1–8. doi: 10.1109/IJCNN48605.2020.9207474
- Japkowicz, N., and Stephen, S. (2002). The class imbalance problem: A systematic study. *Intell. Data Anal.* 6, 429–449.
- Kachuee, M., Fazeli, S., and Sarrafzadeh, M. (2018). "ECG heartbeat classification: A deep transferable representation," in *Proceedings of the IEEE international conference on healthcare informatics*, (Piscataway, NJ: IEEE), 443–444. doi: 10.3390/s21217233
- Kasabov, N. K. (2014). NeuCube: A spiking neural network architecture for mapping, learning and understanding of spatio-temporal brain data. *Neural Netw.* 52, 62–76.
- Kiranyaz, S., Ince, T., and Gabbouj, M. (2016). Real-time patient-specific ECG classification by 1-D convolutional neural networks. *IEEE Trans. Biomed. Eng.* 63, 664–675. doi: 10.1109/TBME.2015.2468589
- Lee, C., Panda, P., Srinivasan, G., and Roy, K. (2018). Training deep spiking convolutional neural networks with stdp-based unsupervised pre-training followed by supervised fine-tuning. *Front. Neurosci.* 12:435. doi: 10.3389/fnins.2018.00435
- Ma, Y., Donati, E., Chen, B., Ren, P., Zheng, N., and Indiveri, G. (2020). "Neuromorphic implementation of a recurrent neural network for EMG classification," in *Proceedings of the 2020 2nd IEEE international conference on artificial intelligence circuits and systems (AICAS)*, (Piscataway, NJ: IEEE), 69–73. doi: 10.1109/AICAS48895.2020.9073810
- Mirowski, P., Madhavan, D., LeCun, Y., and Kuzniecky, R. (2009). Classification of patterns of eeg synchronization for seizure prediction. *Clin. Neurophysiol.* 120, 1927–1940.
- Parajuli, N., Sreenivasan, N., Bifulco, P., Cesarelli, M., Savino, S., Niola, V., et al. (2019). Real-time EMG based pattern recognition control for hand prostheses: A review on existing methods, challenges and future implementation. *Sensors* 19:4596. doi: 10.3390/s19204596
- Pu, D., and Cook, M. (2015). Unsupervised learning of digit recognition using spike-timing-dependent plasticity. *Front. Comput. Neurosci.* 9:99. doi: 10.3389/fncom.2015.00099
- Roy, K., Jaiswal, A., and Panda, P. (2019). Towards spike-based machine intelligence with neuromorphic computing. *Nature* 575, 607–617.
- Rueckauer, B., Lungu, I.-A., Hu, Y., Pfeiffer, M., and Liu, S.-C. (2017). Conversion of continuous-valued deep networks to efficient event-driven networks for image classification. *Front. Neurosci.* 11:682. doi: 10.3389/fnins.2017.00682
- Shoeb, A. H. (2009). *Application of machine learning to epileptic seizure onset detection and treatment*. Ph.D. dissertation. Cambridge, MA: Massachusetts Institute of Technology.
- Shrestha, S. B., and Song, Q. (2018). Robustness to training disturbances in SpikeProp learning. *IEEE Trans. Neural Netw. Learn. Syst.* 29, 3126–3139. doi: 10.1109/TNNLS.2017.2713125
- Simão, M., Neto, P., and Gibaru, O. (2019). EMG-based online classification of gestures with recurrent neural networks. *Pattern Recognit. Lett.* 128, 45–51.
- Srinivasan, G., Sengupta, A., and Roy, K. (2016). Magnetic tunnel junction based long-term short-term stochastic synapse for a spiking neural network with on-chip STDP learning. *Sci. Rep.* 6:29545. doi: 10.1038/srep29545
- Tavanaei, A., Ghodrati, M., Kheradpisheh, S. R., Masquelier, T., and Maida, A. (2019). Deep learning in spiking neural networks. *Neural Netw.* 111, 47–63.
- Tian, F., Yang, J., Zhao, S., and Sawan, M. (2021). "A new neuromorphic computing approach for epileptic seizure prediction," in *Proceedings of the 2021 IEEE international symposium on circuits and systems (ISCAS)*, (Piscataway, NJ: IEEE), 1–5. doi: 10.1109/ISCAS51556.2021.9401560
- Truong, N. D., Nguyen, A. D., Kuhlmann, L., Bonyadi, M. R., Yang, J., Ippolito, S., et al. (2018b). Integer convolutional neural network for seizure detection. *IEEE J. Emerg. Sel. Top. Circuits Syst.* 8, 849–857. doi: 10.1109/JETCAS.2018.2842761
- Truong, N. D., Nguyen, A. D., Kuhlmann, L., Bonyadi, M. R., Yang, J., Ippolito, S., et al. (2018a). Convolutional neural networks for seizure prediction using intracranial and scalp electroencephalogram. *Neural Netw.* 105, 104–111. doi: 10.1016/j.neunet.2018.04.018
- Wei, W., Dai, Q., Wong, Y., Hu, Y., Kankanhalli, M., and Geng, W. (2019). Surface-electromyography-based gesture recognition by multi-view deep learning. *IEEE Trans. Biomed. Eng.* 66, 2964–2973. doi: 10.1109/TBME.2019.2899222
- World Health Organization (2017). *Cardiovascular diseases (CVDs)*. W. H. O. fact sheet 317. Available online at: [https://www.who.int/en/news-room/fact-sheets/detail/cardiovascular-diseases-\(cvds\)](https://www.who.int/en/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)) (accessed September 2019).
- Xu, Y., Yang, J., Zhao, S., Wu, H., and Sawan, M. (2020). "An end-to-end deep learning approach for epileptic seizure prediction," in *Proceedings of the 2020 2nd IEEE international conference on artificial intelligence circuits and systems (AICAS)*, Genova, 266–270.
- Zhang, W., Yu, L., Ye, L., Zhuang, W., and Ma, F. (2018). "ECG signal classification with deep learning for heart disease identification," in *Proceedings of the 2018 international conference on big data and artificial intelligence (BDAI)*, (Piscataway, NJ: IEEE), 47–51. doi: 10.1109/BDAI.2018.8546681
- Zhang, Y., Guo, Y., Yang, P., Chen, W., and Lo, B. (2020). Epilepsy seizure prediction on EEG using common spatial pattern and convolutional neural network. *IEEE J. Biomed. Health Inf.* 24, 465–474. doi: 10.1109/JBHI.2019.2933046
- Zhao, S., Yang, J., Xu, Y., and Sawan, M. (2020). "Binary single-dimensional convolutional neural network for seizure prediction," in *Proceedings of the 2020 IEEE international symposium on circuits and systems (ISCAS)*, Seville, (Piscataway, NJ: IEEE), 1–5.