An Energy-Efficient Graphene-based Spiking Neural Network Architecture for Pattern Recognition

Laurencin, Nicoleta Cucu; Timmermans, Charles; Cotofana, Sorin D.

DOI
10.1109/ISCAS58744.2024.10558243

Publication date
2024

Document Version
Final published version

Published in
ISCAS 2024 - IEEE International Symposium on Circuits and Systems

An Energy-Efficient Graphene-based Spiking Neural Network Architecture for Pattern Recognition

Nicoleta Cucu Laurenciu*, Charles Timmermans*, Sorin D. Cotofana†
*High Energy Physics Department, IMAPP, Radboud University, The Netherlands.
†Quantum and Computer Engineering Department, Delft University of Technology, The Netherlands.
*{N.CucuLaurenciu, C.Timmermans}@science.ru.nl †{S.D.Cotofana}@tudelft.nl

2024 IEEE International Symposium on Circuits and Systems (ISCAS) | 979-8-3503-3099-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/ISCAS58744.2024.10558243

***Abstract*—In this paper we propose a generic graphene-based Spiking Neural Network (SNN) architecture for pattern recognition and the associated weight values initialization methodology. The SNN has a Winner-Takes-All 3-layer structure and exhibits tuneable recognition accuracy by exploiting inter-patterns similarity/dissimilarity. To demonstrate the capabilities of our proposal we present an SNN instance tailored for low resolution MNIST handwritten digits recognition and evaluate its recognition accuracy by means of SPICE simulations. 2 voltage levels are initially utilized for synaptic weight values representation and the recognition accuracy varies from 75.8% to 99.2%, which, together with its compactness and energy efficient (pJ range/spike), suggests that our approach has great potential for edge device implementations.**

***Index Terms*—Graphene, GNR, Nanoelectronics, Neuromorphic Computing, Spiking Neural Network, Pattern Recognition.**

I. Introduction

The proliferation of on-chip battery-powered edge intelligence has driven and in turn was driven by a paradigm shift in neural network architectures to enable energy-effective neuromorphic computing. Spiking Neural Networks (SNNs) that aim to mimic the excellent human brain properties (e.g., massively parallel processing, in-memory computing, ultra-low energy consumption, high resilience and adaptation flexibility) by replicating its structure and operation principles, are the forefront drivers of the neuromorphic computing revolution.

During recent years, many efforts have been made for SNN basic constituents, i.e., neurons and synapses, modelling and implementation by means of complex CMOS circuitry, and more recently by emerging technologies, e.g., memristors [1], phase change memories [2]) in conjunction with CMOS circuitry. However, such approaches are still restricted from the point of view of large-scale energy-efficient implementations. Graphene, one of the prominent post-Si forerunners, exhibits a set of unique properties [3], [4] which makes it particularly attractive for neuromorphic implementations, e.g., ballistic transport, inherently analog operation, and biocompatibility, and previous work demonstrated graphene's suitability for artificial neurons [5], synapses [6], [7], and SNNs [8], [9].

SNNs are fundamentally different from ANNs, from inner structure to operation principle. The SNN event-driven operation and its temporal and local nature of information processing require a rethinking of the architecture design, operation, and training process. ANNs' well established and effective training methods, e.g., back propagation, cannot be utilized for SNNs, because of the discontinuous, non-differentiable nature of spikes generation as well the involved complex dynamics. As such, training is gating the SNNs proliferation and a popular approach to deal with it is to first design the ANN architecture, train it via traditional ANN methods, and subsequently convert the ANN into SNN [10], [11]. However, while bypassing some SNN training difficulties (e.g., gradient computation), this approach fails to capture the neurons and synapses temporal dynamics. Furthermore, conversion imposes a series of constraints on the initial ANN structure that might diminish the SNN accuracy [12]. Very few approaches [13] directly design and train SNNs, and the majority of them are conventional ANN rate-based approximations (e.g., deep networks) [14], and thus while being faster and more energy efficient, no accuracy gains are expected relative to their ANN counterparts.

In this paper, we propose a generic graphene-based SNN architecture for pattern recognition and the associated design methodology for initial synaptic weight values determination. The SNN architecture relies on a Winner-Takes-All structure augmented with inter-pattern similarities/dissimilarities sub-networks. The initial weight values are determined via an off-line analysis of the training set similarities/dissimilarities statistics, thus we rely on a design-time learning and not no a traditional training. As proof-of-concept, we present a graphene SNN architecture instance for a hand-written digit recognition problem for which our SPICE simulations indicate digits recognition accuracies from 75.8% to 99.2%. The main features of the proposed generic SNN architecture can be summarised as: (i) compactness and energy efficiency due to effective graphene-based implementation of neurons and synapses [8], and simple architecture, and (ii) desired recognition accuracy tunable architecture, which are edge device desirable features.

The rest of the paper is organized as follows: Section II presents the generic graphene-based device employed for neurons and synapses implementations. Section III introduces the generic SNN architecture and the synaptic values determination methodology, and presents an SNN instance for a handwritten digits recognition. In Section IV we evaluate the recognition ability by means of SPICE simulations and Section V summarizes our main findings.

II. Background

Figure 1 illustrates the generic graphene-based device that is utilized for implementing spiking neurons and STDP synapses. It relies on a Graphene NanoRibbon (GNR), which acts as

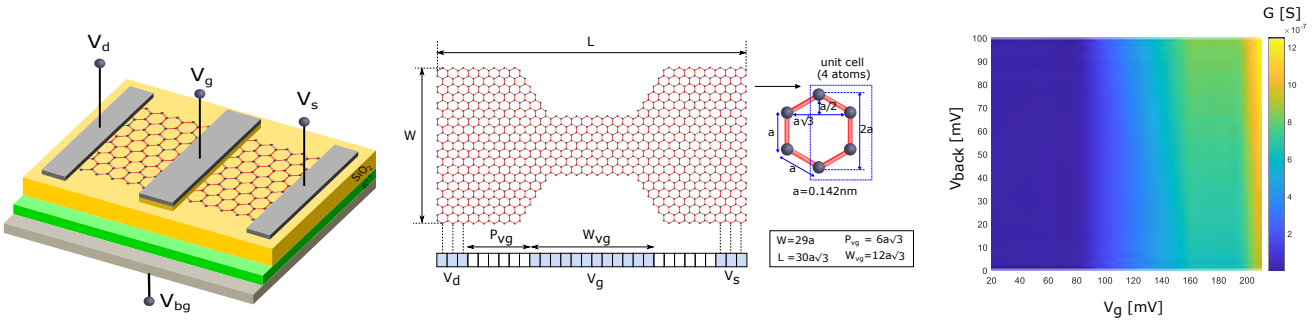

Fig. 1: Generic graphene-based device and conduction map example.

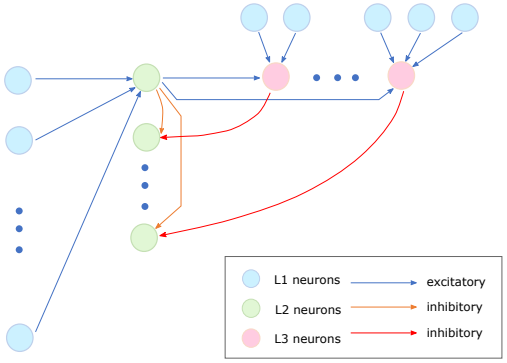

Fig. 2: SNN architecture for pattern recognition.

conduction channel when the device is subjected to a drain-to-source bias voltage. The actual conductance is controlled by external voltages applied on the top and/or bottom gates [15] and the nanoribbon geometry and contact topology determines the device conduction profile [16]. Note that all neurons and synapses utilized in this paper are GNR-based [8].

III. SNN Architecture and Network Dynamics

In this Section, we describe the proposed generic pattern recognition SNN architecture and the afferent design methodology for determining the initial weight values. Then, we present an SNN architecture instance for reduced resolution MNIST digit recognition.

The proposed SNN architecture, illustrated in Figure 2, has an output layer L2 whose neurons are bijectively associated with the to be discriminated categories and relies on a Winner-Takes-All (WTA) strategy, i.e., the L2 neurons mutually inhibit each other, until after some time only one neuron continues to fire, namely the one which label (category) appears to be the most appropriate for the applied input pattern. As WTA neurons input, the typical approach is to exploit similarities between same category patterns. This is reflected by L1 to L2 all-to-all excitatory synaptic connectivity. Nevertheless, this only works for few discrimination categories and for patterns that have a low degree of correlation for different categories, i.e., are easily distinguishable. To this end, we propose to augment the WTA with a subnetwork that exploits pairwise dissimilarities in order to provide additional inhibition for aiding the discrimination of highly correlated categories. Small L1 to L3 excitatory clusters that reflect particular dissimilarities between 2 hard to distinguish categories, can together provide the additional inhibition extent from one category to the other. We note that no learning/training is involved, as the weights are determined based on an off-line analysis according to certain similarity/dissimilarity statistics.

To exemplify the proposed SNN architecture design, we consider a hand-written digits recognition task. Specifically, we have used the UCI handwritten digits dataset [17], which consists of 3823 training and 1797 testing images, where each image is an 8×8 pixels, grayscale image of digits 0 to 9, and where each pixel value is an integer in the range $[0, 16]$. The proposed SNN architecture follows the structure in Figure 2 and can beperceived as consisting of 3 subnetworks: a subnetwork that exploits digits similarity, a subnetwork that exploits digits dissimilarity, and a third Winner-Takes-All subnetwork that carries out the digits discrimination via a competition mechanisms. L1 is the input layer, with 64 neurons each corresponding to an image pixel, L2 is the SNN output layer with 10 neurons, and L3 may include up to 100 neurons to add L2 pairwise digits disambiguation for certain digits (which are more similar to each other and harder to distinguish from one another).

The images are provided as input stimuli to the SNN first layer of 64 sensory neurons, with every pixel corresponding to an L1 spiking neuron. Pixel values are binarized and converted into spike event (for pixel values ≥ 5) and no spike event (for pixel values < 5). Different binarization threshold values were tested experimentally, however no single optimal value was found. Nevertheless, a threshold of 5 did result in higher recognition accuracy for some of the digits that are harder to distinguish because of overlapping features with other digits. We note that, since the images are binarized they can be straightforward converted into a spike-based spatial representation, bypassing the need for random number generators and extra circuitry associated with existing SNN input encoding methods, such as rate encoding, temporal coding, rank order coding, etc. [18], [19].

IV. Simulation Results

As concerns the simulation environment, the preliminary analysis for initial synaptic weights is performed in Matlab, while the actual graphene-based SNN architecture is simulated in SPICE [20].

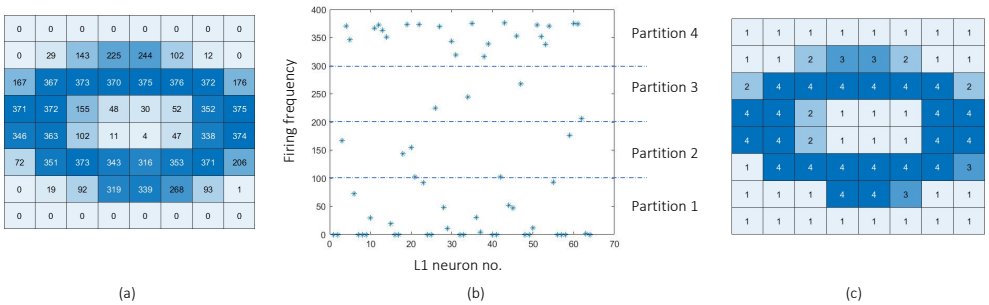

Fig. 3: L1 neurons partitioning example vis-a-vis their firing frequency for all digit 0 training images (a) digit 0 firing frequency stencil, (b) neurons grouping into 4 equidistant partitions, (c) 4 equidistant partitioning stencil.

		Digits Recognition Accuracy (R_{acc}) [%]									
Digit		0	1	2	3	4	5	6	7	8	9
NImgTrain		376	389	380	389	387	376	377	387	380	382
Baseline	[max no. of partitions]	82.2	1.5	12.1	46.5	22.5	11.2	68.7	44.2	98.7	-30.6
2 PARTITIONS	[0 - 200 - 400]	87.3	8.7	16.9	46.5	30.7	18.6	68.7	45.4	99.3	7.6
2 PARTITIONS	[0 - 150 - 400]	86.7	23.9	33.4	-1.8	39.8	-5.9	66.3	33.6	42.1	55.3
3 PARTITIONS	[0 - 200 - 300 - 400]	88.3	-8.5	41.1	74.8	27.9	-6.4	70	53.5	99.6	-1.3
3 PARTITIONS	[0 - 150 - 250 - 400]	88.3	29.6	45.5	70.1	39.5	9.3	69.3	49.4	62.1	28.5
4 PARTITIONS	[0 - 100 - 200 -300 - 400]	88.3	9.3	51.9	74.5	52.7	10.6	69	56.8	99.8	58.9

Fig. 4: Example of L1 neurons partitioning for synaptic weights initialization.

A. Similarity sub-network

A first architectural decision concerns the initial L1 to L2 synaptic weights. In principle each and every synaptic weight could be initialized to a different value. However, since for edge devices area and energy consumption are critical, it would be desirable to reduce as much as possible the cardinality of the initial synaptic weight values set. One solution to determine the optimal number of L1 to L2 synaptic partitions (where to each partition is assigned a same initial weight value), is to rely on the neurons firing patterns (e.g., firing frequency). In particular, we applied all 376 training images belonging to digit 0 and recorded the firing frequency for each of the 64 L1 neurons. Then we partitioned the L1 neurons into multiple groups, based on their firing frequency similarity and subsequently carried out the same analysis independently for all the other digits. Figure 3 graphically illustrates the L1 neurons firing frequencies for the entire set of digit 0 376 training images, and an example of partitioning into 4 groups based on the frequency values similarity.

As performance measure for the similarity sub-network synaptic weights initialization, we consider the SNN recognition accuracy for a digit dig, which is computed as follows:

$$\mathrm{R_{acc}} = (\mathrm{NImgTrain}_{dig} - \mathrm{Nfaulty}_{dig}) * 100 / \mathrm{NImgTrain}_{dig}, \quad (1)$$

where $\mathrm{NImgTrain}_{dig}$ denotes the total number of training images for digit dig, and Nfaulty is the number of false positive recognitions for that digit. To determine the quantization degree of the synaptic weight values, we investigated the effects on the SN recognition accuracy when considering (i) different number of partitions individually for every digit, (ii) different number of partitions common for all digits, and (iii) the firing frequency threshold value for assigning neurons to different partitions when the number of partitions is given. Simulation results, exemplified in Figure 4, indicated that (i) in general as the number of partitions increases, the recognition accuracy improves for some digits, but degrades for other digits, and after a certain number of partitions, there is an overall degradation for all digits recognition accuracy, and (ii) the accuracy is very much dependent on finding the optimal thresholds to separate the partitions. All results in Figure 4 are reported relative to the baseline configuration for which every $L1$ neuron forms its own partition. For the 2 partitions case, setting the firing frequency threshold to 150 instead of 200 yields better accuracy ranging from 4.5% up to 85.9% for digits $\{0; 1; 2; 4; 9\}$, while degrading the accuracy ranging from -2.4% up to -56.6% for digits $\{3; 5; 6; 7; 8\}$. Thus, for 2 partitions, a threshold of 200 leads to better results overall. Similar conclusion can be drawn from the 3 partitions example. We note that the SNN architectural design for the digits similarity subnetwork should be done in tandem with the digits dissimilarity subnetwork, as otherwise would misguide the design space exploration and lead to suboptimal results. Furthermore, for determining the partitions Pareto optimal configuration, several aspects require consideration, e.g., a higher recognition accuracy for digits which pose recognition difficulty because of their high degree of similarity with other digits (e.g., 6 and 8) is preferable to a higher accuracy percentage for digits which are more easily distinguishable, does a higher number of partitions justifies the extra area and energy circuitry associated with multiple voltage levels or a lower number of partitions in tandem with increasing complexity of the digits dissimilarity subnetwork can resolve recognition disambiguation more cost effectively, etc. From the design space exploration it was found that 2 equidistant partitions not only offered the best area and energy trade-offs, but also outperformed recognition accuracy wise configurations with multiple partitions.

B. Dissimilarity sub-network

The dissimilarity subnetwork is designed for pairwise digit inhibition and only for the digits for which the SNN recognition accuracy is low. Figure 5 exemplifies the design principle of the dissimilarity subnetwork from digit 0 to digit 1. To derive the L1 neuronal cluster, the 2 digit stencils can be subtracted and the neurons with the lowest and/or highest firing activity which correspond to the most dissimilar features present or absent from one digit, are selected. Depending on the desired SNN recognition accuracy more or less neurons can be included and grouped either as a single cluster or

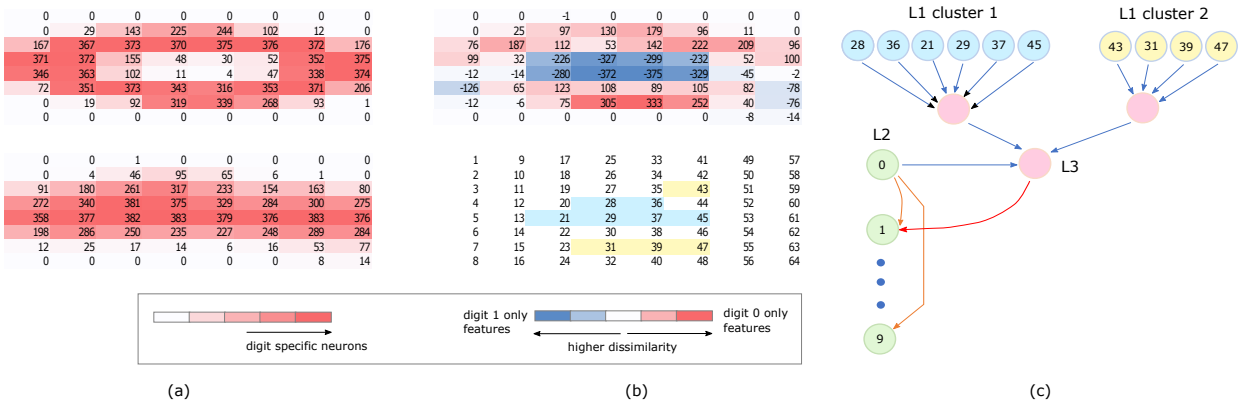

Fig. 5: Inhibition from digit 0 to digit 1: (a) digit 0 and digit 1 L1 stencils, (b) digit 0 to digit 1 dissimilarity stencil, and (c) example of 2 L1 dissimilarity clusters.

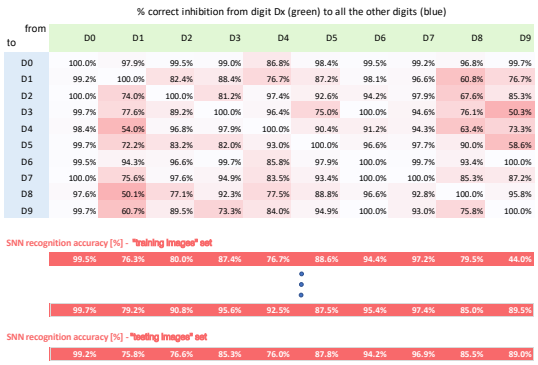

Fig. 6: Pairwise digit inhibition and SNN recognition accuracy for the set of training and testing images.

as multiple clusters as illustrated in Figure 5 (c). Figure 6 exemplifies numerically the pairwise inhibition effects of the dissimilarity subnetwork for all digits. The % correct inhibition from digit A to every other digit B is derived as the ratio between the number of A digit inhibitions when applying as SNN input B images and the total number of train images for digit A. Suppose there are 2 digits - A and B, and we would like to determine the L1 neuronal cluster for inhibition from digit A to B. Ideally, the neuronal cluster should reflect features that are present in all images from digit A (otherwise less images from digit B get inhibited) and at the same time absent in all images from digit B (otherwise there are more false positive inhibitions). In practice this trade-off has to carefully considered, since it affects the recognition ability not only for the considered 2 digits, but for the other digits as well. To this end, we define a cost function for each L1 neuron, which is increased when the neuron benefits overall the entire SNN, and decreased otherwise. Simulation results in Figure 6 indicate that the inhibition is effective for some digits (% correct inhibition is higher), which might rely solely on the WTA L2 inhibition and not require any inhibition via the dissimilarity sub-network. On the contrary, for other digits the % correct inhibition is lower but can be increased more or less depending on the SNN targeted recognition accuracy, for instance by adding more L1 neuronal clusters in those particular cases. Figure 6 shows the SNN recognition accuracy for the set of training images, and for the set of testing images. The accuracy is successively increased by improving the dissimilarity sub-network for the digits with lower % of correct inhibition. We note that the SNN recognition accuracy for the training and testing set of images is quite similar with the exception of digit 2, for which the accuracy degraded with 16%, and that a simple SNN structure with 2-voltage levels only for synaptic initialization can achieve good performance. We note that this performance can be further improved via extending the dissimilarity sub-network if very high accuracy is the target. We note that it is hard to make comparison for neural network architectures. For instance, comparing with an ANN is rather irrelevant, since SNNs and ANNs are fundamentally different from multiple standpoints, e.g., structure, mode of operation, different design spaces, different algorithms, etc. Furthermore, comparing against other spiking neural networks is also less relevant since the vast majority of them rely on deep neural network structure with hundreds of parameters and multiple layers, while in our case the SNN architecture has a simple structure, which would essentially translate to orders of magnitude savings. As concerns the energy expenditure, for both graphene-based neuron and synapse the energy/spike is in the order of 10^{-7} to 10^{1} pJ, for ps to ms spike timescale [8], which indicates the energy effectiveness of proposed SNN structure.

V. Conclusions

In this paper we proposed a generic graphene-based SNN architecture for pattern recognition and a methodology for initial weight values determination. The main advantages of proposed SNN architecture are: (i) simple architecture, (ii) energy effectiveness due to the simple architecture and the utilization of graphene neurons and synapses, and (iii) recognition accuracy aware design methodology. We also presented a proof-of-concept SNN architecture for handwritten digits recognition. SPICE simulation results indicate that a simply binary weighted synaptic SNN network can achieve good recognition accuracy performance (up to 99.2%) and thus eliminate the recourse to deep neural network or complicated design with hundreds of parameters and layers.

REFERENCES

[1] Strukov, D. B., Snider, G. S., Stewart, D. R., and Williams, R. S., "The missing memristor found." *Nature*, vol. 453, no. 7191, p. 80, 2008. [Online]. Available: https://doi.org/10.1038/nature06932

[2] Philip Wong, H.-S., et al., "Phase Change Memory." *Proceedings of the IEEE*, vol. 98, no. 12, pp. 2201–2227, 2021. [Online]. Available: https://doi.org/10.1109/JPROC.2010.2070050

[3] Avouris, P., and Dimitrakopoulos, C., "Graphene: synthesis and applications." *Materials Today*, vol. 15, no. 3, pp. 86–97, 2012.

[4] Allen, M. J., Tung, V. C., and Kaner, R. B., "Honeycomb carbon: a review of graphene." *Chemical Reviews*, vol. 110, no. 1, pp. 132–145, 2009.

[5] Wang, H., Cucu Laurenciu, N., Jiang, Y., and Cotofana, S. D., "Ultracompact, entirely graphene-based nonlinear leaky integrate-and-fire spiking neuron." in *IEEE INternational Conference on Circuits and Systems (ISCAS)*, 2020, pp. 1–5. [Online]. Available: https://doi.org/10.1109/ISCAS45731.2020.9181092

[6] ——, "Graphene-Based Artificial Synapses with Tunable Plasticity." *ACM Journal on Emerging Technologies in Computing Systems*, vol. 17, no. 4, pp. 1–21, 2021. [Online]. Available: https://doi.org/10.1145/3447778

[7] Cucu Laurenciu, N., Timmermans, C., and Cotofana, S. D., "Low Energy, Non-Cortical, Graphene Nanoribbon-Based STDP Plastic Synapses." *IEEE Nanotechnology Magazine*, vol. 16, no. 6, pp. 4–13, 2022. [Online]. Available: https://doi.org/10.1109/MNANO.2022.3208722

[8] Wang, H., Cucu Laurenciu, N., and Cotofana, S. D., "A Reconfigurable Graphene-Based Spiking Neural Network Architecture." *IEEE Open Journal of Nanotechnology*, vol. 2, pp. 59–71, 2021. [Online]. Available: https://doi.org/10.1109/OJNANO.2021.3094761

[9] Wang, H., Cucu Laurenciu, N., Jiang, Y., and Cotofana, S. D., "Compact Graphene-Based Spiking Neural Network With Unsupervised Learning Capabilities." *IEEE Open Journal on Nanotechnology*, vol. 1, pp. 135–144, 2020. [Online]. Available: https://doi.org/10.1109/OJNANO.2020.3041198

[10] Merolla, P., Arthur, J., Akopyan, F., Imam, N., Manohar, R., and Modha, D. S., "A digital neurosynaptic core using embedded crossbar memory with 45pj per spike in 45nm." in *IEEE Custom Integrated Circuits Conference (CICC)*, 2011, pp. 1–4. [Online]. Available: https://doi.org/10.1109/CICC.2011.6055294

[11] Hussain, S., Liu, S.-C., and Basu, A., "Improved margin multi-class classification using dendritic neurons with morphological learning." in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2014, pp. 2640–2643. [Online]. Available: https://doi.org/10.1109/ISCAS.2014.6865715

[12] Gao, H., et al., "High-Accuracy Deep ANN-to-SNN Conversion using Quantization-Aware Training Framework and Calcium-Gated Bipolar Leaky Integrate and Fire Neuron." *Frontiers in Neuroscience*, vol. 17, pp. 1–11, 2023. [Online]. Available: https://doi.org/10.3389/fnins.2023.1141701

[13] Wu, G., Liang, D., Luan, S., and Wang, J., "Training Spiking Neural Networks for Reinforcement Learning Tasks With Temporal Coding Method." *Frontiers in Neuroscience*, vol. 16, pp. 1–11, 2022. [Online]. Available: https://doi.org/10.3389/fnins.2022.877701

[14] Lee, J. H., Delbruck, T., and Pfeiffer, M., "Training Deep Spiking Neural Networks Using Backpropagation." *Frontiers in Neuroscience*, vol. 10, pp. 1–13, 2016. [Online]. Available: https://doi.org/10.3389/fnins.2016.00508

[15] Cotofana, S. D., Dimitrakis, P., Enachescu, M., Karafyllidis, I., Rubio, A., and Sirakoulis, G. C., "On graphene nanoribbon-based nanoelectronic circuits viability." in *42nd Workshop Compound Semicond. Devices Integr. Circuits Held Eur.*, 2018, pp. 35–36.

[16] Jiang, Y., Cucu Laurenciu, N., and Cotofana, S. D., "On Basic Boolean Function Graphene Nanoribbon Conductance Mapping." *IEEE Transactions on Circuits and Systems-I: Regular Papers.*, vol. 66, no. 5, pp. 1948–1959, 2018. [Online]. Available: https://doi.org/10.1109/TCSI.2018.2882310

[17] UCI Machine Learning Repository. Irvine, CA: University of California [Online]. Available: https://archive.ics.uci.edu/ml/datasets/optical+recognition+of+handwritten+digits.

[18] Q. Wu, M. McGinnity, L. Maguire, B. Glackin, and A. Belatreche, *Learning Mechanisms in Networks of Spiking Neurons*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 171–197. [Online]. Available: https://doi.org/10.1007/978-3-540-36122-0_7

[19] Guo, W., Fouda, M. E., Eltawil, A. M., and Salama, K. N., "Neural Coding in Spiking Neural Networks: A Comparative Study for Robust Neuromorphic Systems." *Frontiers in Neuroscience*, vol. 15, pp. 1–21, 2021. [Online]. Available: https://doi.org/10.3389/fnins.2021.638474

[20] Cadence. [Online]. Available: https://www.cadence.com/.