Walters, Ben, Lammie, Corey, Yang, Shuangming, Jacob, Mohan V., and Rahimi Azghadi, Mostafa (2024) ***Unsupervised character recognition with graphene memristive synapses.*** **Neural Computing and Applications, 36 pp. 1569-1584.**

Unsupervised Character Recognition with Graphene Memristive Synapses

Ben Walters[1], Corey Lammie[1], Shuangming Yang[2], Mohan V Jacob[1] and Mostafa Rahimi Azghadi[1*]

[1*]College of Science and Engineering, James Cook University, 1 James Cook Drive, Townsville, 4811, QLD, Australia.
[2]School of Electrical and Information Engineering, Tianjin University, China.

*Corresponding author(s). E-mail(s): mostafa.rahimiazghadi@jcu.edu.au;
Contributing authors: ben.walters@my.jcu.edu.au; corey.lammie@jcu.edu.au; yangshuangming@tju.edu.cn; mohan.jacob@jcu.edu.au;

Abstract

Memristive devices being applied in neuromorphic computing are envisioned to significantly improve the power consumption and speed of future computing platforms. The materials used to fabricate such devices will play a significant role in their viability.Graphene is a promising material, with superb electrical properties and the ability to be produced in large volumes. In this paper, we demonstrate that a graphene-based memristive device could potentially be used as synapses within Spiking Neural Networks (SNNs) to realise Spike Timing Dependant Plasticity (STDP) for unsupervised learning in an efficient manner. Specifically, we verify the operation of two SNN architectures tasked for single-digit (0-9) classification: (i) a single layer network, where inputs are presented in $\mathbf{5 \times 5}$ pixel resolution, and (ii) a larger network capable of classifying the Modified National Institute of Standards and Technology (MNIST) dataset. Our work presents the first investigation and large-scale simulation of the use of graphene memristive devices to perform a complex pattern classification task. In favour of reproducible research, we will make our code and data publicly available.*

* https://anonymous.4open.science/r/SNN-GrapheneSynapses-2E0D

This can pave the way for future research in using graphene devices with memristive capabilities in neuromorphic computing architectures.

Keywords: Unsupervised Learning, SNN, Graphene, Memristors, Simulation

1 Introduction

There is a plethora of work dedicated to the memristor, ever since its concept was proposed by Leon Chua in [1]. Specifically, the hysteresis effect observed in memristive devices allows for in-situ learning and memory, breaking down the Von-Neumann bottleneck present in traditional computing architectures [2, 3]. The link between a resistive switching device and memristors was first demonstrated by HP labs in 2008 [4], and ever since many different devices have been created, with a variety of switching properties observed. Examples include the formation of conductive filaments [5], Phase Change Memory (PCM) [6], altering magnetic [7] and ferroelectric [8] polarisations, and interfacial switching [9] devices.

One of the most important aspects of memristor design is choosing its constructive material, with different materials such as conventional oxides [10], and 2D materials like graphene [11], chalcogenides [12] and others [13] being recent common choices. Of particular interest in this paper is the use of graphene, not only due to its superior properties, i.e. high conductivity at room temperature, mechanical robustness and atomically thin size, but also due to the ability to manufacture graphene sustainably and at a low cost [14]. Since memristors have been demonstrated to be able to play a significant role in Artificial Intelligence (AI) hardware [3, 15–19], this can be beneficial in building large-scale neuromorphic, Machine Learning (ML), and Deep Learning (DL) systems towards green intelligent machines.

There are many forms of ML and DL architectures including Deep Neural Networks (DNNs) [20, 21], which are capable of performing many challenging engineering tasks [22]. These architectures, namely second generation Artificial Neural Networks (ANNs), usually learn by minimising an error function through optimising their learning parameters, which may differ from the learning processes in biological brains. On the other hand, neuromorphic and bio-inspired computing systems, emulate biophysical processes that occur in the brain, which are believed to be responsible for biological learning and memory. SNNs, commonly referred to as the third generation of ANNs, are the fundamental architecture of neuromorphic systems. They transmit information in the form of voltage spikes [23], asynchronously. Consequently, they are capable of efficient low-power operation [24], because unlike conventional ANNs that work on a synchronous clock-driven basis, they process data only when available and are in a low-power state at other times.

Another difference between neuromorphic SNNs and their DNN counterparts is the way they learn. Although many SNNs nowadays are capable of

Table 1: Related work for unsupervised STDP-based MNIST classification.

Paper	Synapse Learning Rule	No. Epochs	Image Pre-Processing	Memristive Material	Open-Source Simulation
[31]	Weight Dependant STDP	1,3,7,15	Frequency Proportional Spike Trains	N/A	✓
[32]	Simulated memristive	1	Frequency Proportional Spike Trains	N/A	×
[33]	Fixed STDP	3	Frequency Proportional Spike Trains	N/A	×
[34]	Device conductance mapping, STDP	1	Frequency Proportional Spike Trains	Carbon nanotube	×
[35]	Device conductance mapping, STDP	3	Frequency Proportional Spike Trains	$Al/Al_2O_3/Nb_xO_y/Au$	×
[36]	Soft-bound STDP	1	Frequency Proportional Spike Trains	TaO_y/HfO_x	×
[37]	Device conductance mapping, STDP	10	Image thresholding	$Cu/SiO_2/W$	×
[38]	Multi-memristive synapse	3	Frequency Proportional Spike Trains	$Ge_2Sb_2Te_5$	×
[39]	Simulated memristive	Not Specified	Linearly Scaled Activity Patterns	N/A	×
This Paper	**Device current mapping, STDP**	**1,1,2,3,3**	**Image thresholding**	$\mathbf{Al/C_{22}H_{14}/C_{140}H_{42}O_{20}/SiO_2/Si}$	✔

being trained in similar ways to their DNN counterparts, i.e., by using error backpropagation and surrogate gradient descent [25–27] algorithms, many SNNs are trained using biologically inspired learning algorithms. One of the most widely-used and -studied brain-inspired learning mechanisms that has been the subject of myriad studies [3, 28, 29] is STDP. STDP governs the synaptic weight changes in an unsupervised manner, based on the timing of neuronal activity events [30]. When a pre-synaptic potential precedes a post-synaptic potential, potentiation occurs. Depression occurs when the opposite is true.

To fully exploit the remarkable benefits of graphene devices in neuromorphic computing, in this paper, we investigate the potential use of a graphene device to implement STDP. To that end, we first generate a STDP learning mechanism utilising the current-voltage data obtained from sample graphene-pentacene ($Al/C_{22}H_{14}/C_{140}H_{42}O_{20}/SiO_2/Si$) memristive devices, previously fabricated by our group [14]. We then use this graphene-based STDP learning to perform comprehensive simulations of SNNs containing Leaky Integrate and Fire (LIF) neurons with adaptive voltage thresholds for two classification tasks; binary pattern classification and hand-written digit classification.

Our specific contributions are as follows:

1. We generate a STDP window using the I-V data of graphene-pentacene memristive devices for use in the large-scale simulation of SNNs;
2. We evaluate the effectiveness of our graphene-pentacene memristive devices when used in conjunction with LIF neurons with an adaptive threshold voltage using a simple unsupervised pattern classification task by performing extensive simulations, as a preliminary step prior to large-scale simulations;
3. We demonstrate competitive performance on the unsupervised learning of MNIST dataset when simulating larger networks, and report state-of-the-art performance for some network configurations. We provide the code used to obtain these simulation results[1].

It is worth noting that, the utilised graphene-based memristor only presents a sample device. Our open-source model and simulation framework can be used to capture the behaviour of any other memristive devices to use in large-scale neuromorphic simulations.

The rest of the paper is structured as follows. Section 2 describes related work. Section 3 discusses how the graphene device that was used in the network simulations was fabricated and simulated. Section 4 explains the layout and components of a single layer network used to verify the learning in the graphene device using a simple binary pattern classification task. Section 5 presents the network used to perform the MNIST classification task and discusses its achieved results comparable to literature. Finally, the paper is concluded in Section 6.

2 Related Work

2.1 Unsupervised STDP-based MNIST Classification

Many works have investigated unsupervised learning in SNNs. In [31], Diehl and Cook designed a network to perform unsupervised learning of the MNIST dataset. They achieved a classification accuracy of 95.0% with 6,400 excitatory neurons. They also reported simulation accuracies between 80.0% and 90.0% when they tested their network with 100-1600 excitatory neurons. Although 95.0% is one of the highest accuracies using unsupervised STDP learning reported in the literature, Diehl and Cook utilised a non-memristive software-based synapse model.

Brivio *et al.* [32] also performed unsupervised STDP learning of the MNIST dataset using a SNN. Whilst their reported accuracies tended to saturate at 85.0%, their aim was to test architectures that could possibly be implemented in hardware using hybrid Complementary Metal-Oxide-Semiconductor (CMOS)-memristive technologies, unlike [31], which focused on theoretical architectures and learning rules. In another work, Querlioz *et al.* [33] performed unsupervised learning with memristive STDP synapses, and achieved a simulation accuracy of 93.5% with 300 output neurons. Several other works [34, 40–43] have also tested SNNs, using memristive synapses. Table 1 lists the related memristive STDP studies, along with their memristive material choices (where applicable), their customised synaptic learning rule, their image preprocessing techniques, as well as the number of training epochs they have used. We have also listed our graphene-based memristive STDP synapse model.

These works are used as a benchmark for the results of this paper. It is worth noting that, we could not find any previous research that used graphene-based memristors for learning MNIST by an SNN. At the same time, apart from [31] that is a non-memristive simulation study, our work presents the only open-source simulation investigation of the memristive classification of MNIST using STDP learning.

2.2 Graphene-based Synapses in SNNs

Graphene has long been researched and studied due to its widely useful electrical and mechanical properties, such as zero bandgap, linear energy dispersion near the dirac point and a high electron mobility of 15,000 $cm^2V^{-1}s^{-1}$ [44] as well as high thermal conductivity and mechanical robustness [44, 45]. Furthermore, some studies have demonstrated the viability of commercial fabrication of graphene. In particular, [14] has demonstrated that graphene can be fabricated in a rather sustainable manner by natural extracts as precursors. Because of this, graphene is touted to having many potential applications such as transistors, Transparent Conducting Electrodes (TCE's), Photovoltaics, Light-Emitting Diodes (LED's), among others [44]. Hence, many devices ranging in purpose have been fabricated by utilising graphene, such as graphene memristive devices.

Many different types of graphene memristive devices have been developed, such as graphene based transistors [46–48], graphene-oxide based memristors [49–52] as well as other graphene-based memristive devices [14, 53, 54]. Other 2D materials, such as black phosphorous, boron nitrides, dichalcogenides, and perovskite, have picked up interest in this field as well [11, 55–57].

During the process of device characterisation, it is natural to investigate the memristive properties for their suitability within neuromorphic applications. One of the most widely investigated aspects of memristors is their ability to perform the STDP learning rule. The STDP learning rule is a biologically derived learning rule which relates the timing of pre- and post-synaptic voltage spikes to determine changes in synaptic weights [58]. Works such as [59–61] have developed STDP windows for their devices.

In some of these works, graphene-based devices were implemented into various types of Neural Networks (NNs). Some works, such as [49, 50], have implemented their devices into ANNs to perform image recognition, whilst other works such as [46, 60–62], implemented their devices into SNNs. In [46], the network was trained using a supervised learning rule. [61] showed a graphene-based synapse with synaptic plasticity, but did not perform a pattern classification task. However, [62] performed unsupervised learning in a manner very similar to this paper, as did [60] and [63]. In all cases, alphabetic characters were classified, making it difficult to benchmark their performance against other SNNs that have generally been tested on classifying the MNIST dataset. In fact, Table 1 shows that, to the best of our knowledge, no previous work has used graphene-based synapses to perform unsupervised classification of MNIST dataset using the STDP rule, which is performed in this paper.

There have also been studies based on developing memristive models for implementation in SNNs. Some models, such as the Simmons tunnel barrier model, the ion drift models (linear and non-linear), and Stanford-PKU model are complex, physics-based models [64, 65]. Other models such as the Voltage Threshold Adaptive Memristor (VTEAM) model [66], or the Data-Driven Verilog-A model proposed in [67], are empirical. There has even been research investigating analytical models for graphene oxide devices [59]. The

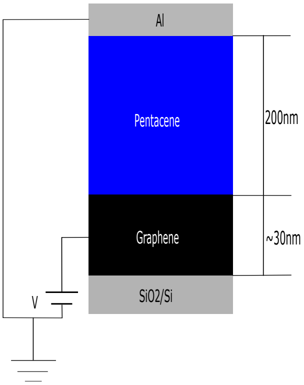


Figure 1: Structure and dimensions of our graphene/pentacene memristive device.

choice of model used is dependent on the properties and switching characteristics of the device in question, and it is sometimes necessary to accommodate for the differences between model and device performance.

3 Graphene Device Implementation

We have previously characterised the memristive behaviour of our graphene-based device [14] used in this work. The device consists of graphene sandwiched in between layers of pentacene, aluminium and fused silica, as shown in Fig. 1. Graphene was fabricated using the Plasma Enhanced Chemical Vapour Deposition (PECVD) method described in [14]. Fig. 2(a) shows the experimental I-V characteristics of two sample fabricated and characterised devices. These memristive devices implemented graphene as an electrode, due to its superior conductivity. The cause of the resistive switching in the device has been previously studied in [68]. In this work, it is determined, through Charge Modulation Spectroscopy (CMS) and Electric Field Induced Second Harmonic Generation (EFISHG), that hole injection from the graphene pentacene contact was the main contributor to the resistive switching observed within a very similar device. This particular device was chosen due to the methods used to fabricate it. Specifically, the graphene in this device was fabricated using natural precursors [14]. Consequently, the fabrication of the device is considered highly sustainable.

Initially, the VTEAM model [66] was curve-fitted to the averaged reference device I-V (Fig. 2(b)), by minimising the error function specified in [66]. However, it was found that this model could not account for the transitions that occur within the devices. Long-Term Potentiation (LTP) and Long-Term Depression (LTD) dynamics were not used to fit the model, as analog wave-forms with varying periods are used to elicit changes in synaptic weight. Upon further investigation, it was revealed that the I-V data was undergoing significant conductance changes when the voltage was beyond the positive and negative

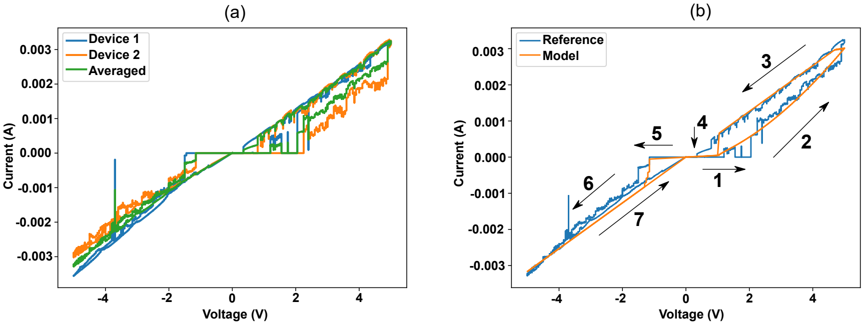


Figure 2: (a) Sample fabricated and characterised graphene memristors used to construct our device model. (b) The averaged reference device data and the model output based on this data, showing the different steps of device IV characteristics.

thresholds, like the VTEAM model, but also when the voltage was reset back to zero. It was therefore decided that the VTEAM memristor model could not be used, hence another approach was devised to utilise the memristive behaviour of our graphene-based device to implement STDP for our unsupervised pattern classification task. This method is described in Section 4.2. Thus, we aimed to implement a new model that, whilst similar to the VTEAM model, accounts for these extra transitions.

The new model has the same switching characteristics as the VTEAM model, however, two points where weight update occurs have been added. Hence, four transition points have been included, labelled “VSetP”, ”VSetN”, “VResetP” and “VResetN”. The “Set” transition points are where the conductance of the device is increased when the magnitude of the voltage applied surpasses these values. Similarly, when the magnitude of the voltage is below the “Reset” points, the device conductance is decreased. The equations used for these models are exactly the same as the ones used in the original VTEAM model, but four equations instead of two are used to account for all of the transitions. The choice of parameters was optimised using Equation 9 discussed in [66].

4 Unsupervised Binary Pattern Classification

In this section, we discuss the structure and building blocks of a simple single-layer network we used in our simulations to perform unsupervised pattern classification using our graphene-based memristive synapse. This represents a preliminary step to better understand the memristive learning dynamics, prior to large-scale simulations.

4.1 Neuron Model

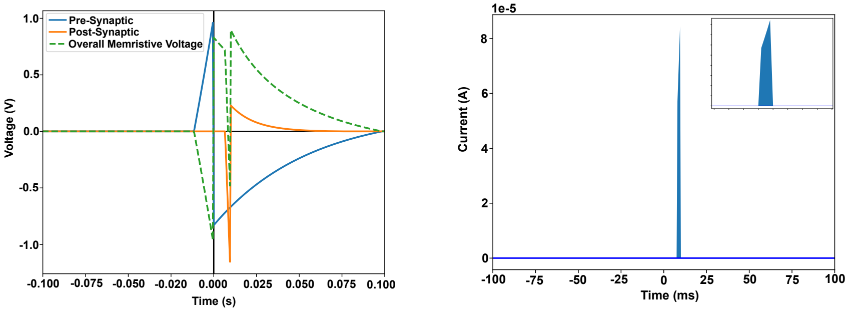
The LIF neuron was used, as it has previously demonstrated significant performance in many neuromorphic systems [3, 69], and it has a much smaller hardware footprint comparable to more complex neuron models, such as the Izhikevich [70] and Hodgkin-Huxley neuron model [71]. The membrane voltage, V, of a LIF neuron is described as

$$\tau \frac{dV}{dt} = (E_{rest} - V) + I_{sum}R, \tag{1}$$

where E_{rest} is the resting potential, τ is the membrane time constant, R is the membrane resistance, and I_{sum} is the sum of the input currents into the neuron [72]. When the membrane potential exceeds the neurons threshold voltage, V_{th}, the neuron's membrane potential resets to its resting potential, E_{rest}. The neuron parameters used in our simulations are as follows: $\tau = 0.0016s$, $R = 500\Omega$ and $E_{rest} = 0.0095V$.

4.2 Synapse Implementation

To generate the STDP windows, various voltages (which were dependant on the timings of pre- and post-synaptic signals) were applied to the model of the device to generate a corresponding current. The values of current for which the voltage exceed either "Set" points were then integrated with respect to time to produce a value of integrated current, which corresponded to a value of ΔT. For the simulations that followed, this value of integrated current was interpreted as the change in current in response to a timing difference between



(a)

(b)

Figure 3: Deriving the STDP waveform. (a) The pre- (blue) and post-synaptic (orange) action potentials for a ΔT of +10ms, summing to an overall memristive voltage waveform (green). (b) The current waveform produced by the pre- and post-synaptic action potentials in (a) which was calculated using the modified VTEAM model, and its close-up (inset).

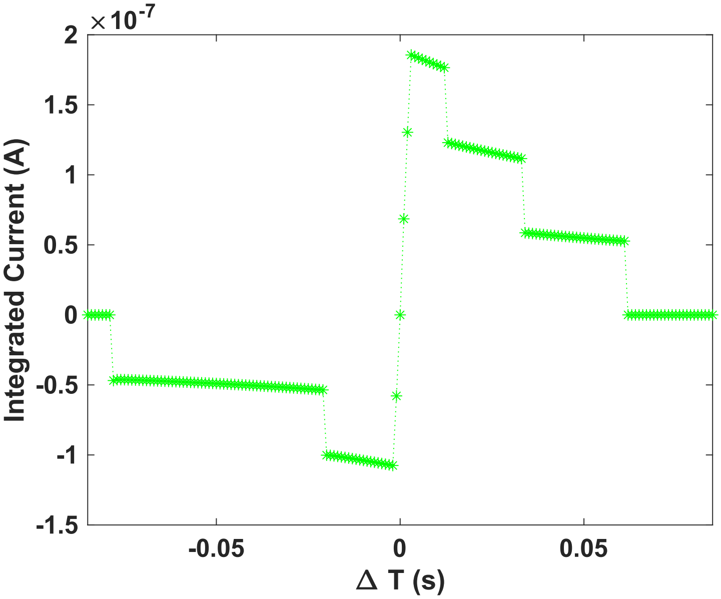


Figure 4: The STDP window of the graphene memristive device, which shows the value of the integrated device current in response to the timing difference between pre- and post-synaptic spikes, i.e., $\Delta T = T_{post} - T_{pre}$.

pre- and post-synaptic signals. The blue shaded area in Fig. 3b was calculated, by integrating current with respect to time. This change in the memristor current in response to ΔT is reminiscent of the change in Excitatory Post Synaptic Current (EPSC) amplitude, which in experimental studies represents the synaptic weight to induce current to its afferent neurons [30]. This way, an STDP window, where the synaptic weight is represented by EPSC, was produced.

In order to produce a viable STDP waveform, the shape of the action potential was carefully considered. Similarly to previous works [73], an exponential shape was chosen as shown in Fig. 3a. By modifying the parameters of this action potential, such as the timing constants, widths and peak voltages for both pre- and post-synaptic spikes, as well as the ratio between pre- and post-synaptic signals, different STDP windows were produced. Next, by implementing several different windows into our developed SNN, its ability to perform a binary pattern classification task was evaluated. This eventually led to the STDP window shown in Fig. 4, which demonstrated the best convergence results for the targeted pattern classification task, using the network architecture described in Fig. 5.

4.3 Network Architecture

The network outlined in [74] had been previously used to perform unsupervised character recognition using the STDP learning rule. As such, a similar network was developed in order to test the graphene memristive synapses.

The single layer network, shown in Fig. 5, had 10 separate, 5×5 binary images presented to it. These binary images were encoded into Poissonian spike trains, where "active" or "on" pixels were encoded with a frequency of 200 Hz and "non-active" or "off" pixels were encoded with a frequency of 20 Hz. Poissonian spike trains were used as these have been demonstrated in previous works such as [31, 33, 36–38] that have performed the same task. These frequencies were chosen after several trials to determine the frequencies which would best result in potentiation and depression. The image patterns were basic, because the purpose of this network was to determine whether or not any form of unsupervised learning could be performed using the proposed graphene memristive synapses.

With the pre-synaptic spike trains temporally encoded, they were applied through excitatory graphene memristive synapses to the output LIF neurons that generated post-synaptic spike trains. The timing of these post- and pre-synaptic spikes were then used to calculate the change in the EPSC of the graphene memristive synapses based on the STDP figure shown in Fig. 4.

To introduce competitive Hebbian learning between neurons [75], as shown in Fig. 5, inhibitory synapses were connected in between all output neurons to activate lateral inhibition. The weight (current) of these synapses were kept fixed at -60 μA. This was found to be a larger current compared to the maximum excitatory current of ± 10 μA for our graphene memristive synapses. Our experiments showed that, the positive feedback current required by the inhibitory synapses need to be significantly larger to suppress the sum of the inputs to specific neurons.

After applying the patterns to the network, the neuronal firing rates as well as the synaptic receptive field of output neurons were monitored to see which neurons converged to which input patterns. These were also monitored to implement homeostasis into the network.

4.4 Homeostatic Plasticity

For certain input patterns, the output neuron associated with that pattern may have a higher or lower neuronal firing rate than other neurons in the network. With lateral inhibition in the network, the difference in neuronal activity is widened, and can lead to single neurons dominating for multiple input patterns, whilst other neurons are unable to converge to any pattern [33]. Consequently, there was a need to regulate the neuronal activity of each neuron so that low-spiking patterns were not overlooked in the training process. Monitoring and adjusting the neuronal activity can be done in many ways, so it is ideal to look for biologically plausible solutions. By searching for biologically plausible solutions, we aim to implement homeostatic mechanisms that are

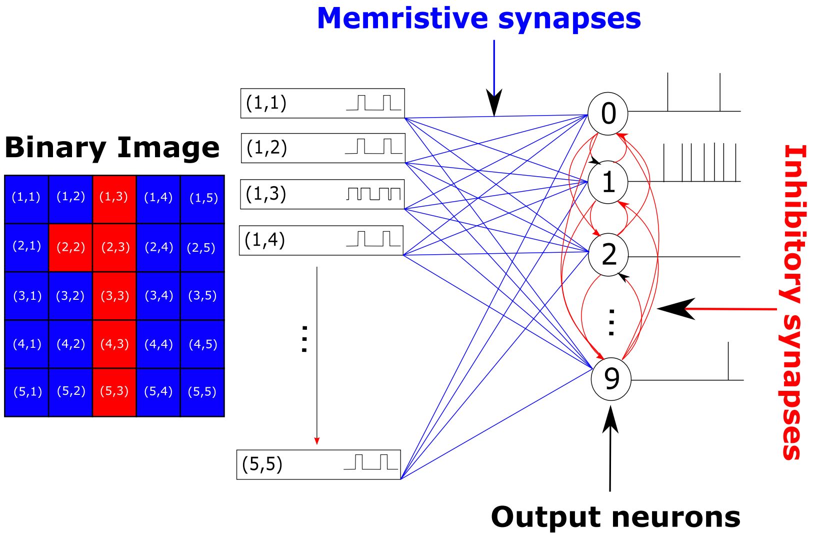


Figure 5: Network architecture used to perform unsupervised character recognition.

more energy efficient, spike-oriented and require less hardware than classical machine learning mechanisms. Plausible mechanisms include synaptic homeostatic regulation and intrinsic neuronal homeostatic regulation [76]. Synaptic homeostatic regulation refers to the strengthening or weakening of synaptic weights, whereas intrinsic homeostatic plasticity refers to the regulating of the neuron's threshold to uphold a certain level of activity [76].

When researching homeostasis in regards to SNNs performing pattern classification tasks, little was found in regards to synaptic homeostatic regulation, likely due to the fact that the learning rules associated with these networks would also modify the synaptic weights. Hence, intrinsic neuronal regulation was used to implement homeostasis. Many different forms of intrinsic homeostatic regulation exist, such as those described in [77] and [78]. However, whilst biologically plausible, these aforementioned models were never used to perform pattern classification. Here, we employed the homeostatic regulation model used in [33] and [35], not only due to its biologically plausible nature, but also because it has demonstrated to significantly improve performance results [33]. This model updates the threshold of the neuron using

$$\frac{dV_{th}}{dt} = \gamma(A - T), \tag{2}$$

where V_{th} is the neuron's threshold, γ is a moderating constant, A is the neuronal firing activity (i.e, the number of times a neuron has fired within a given time period), and T is the target activity for the given period [33]. For our single-layer pattern classification network, a target value of 16 and $\gamma = 0.029$ were chosen. The maximum and minimum thresholds were set between values of $0.010V$ and $0.030V$.

4.5 Training and Classification

Training the single-layer network involved presenting each of the 10 inputs (shuffled in a randomised order) to the network for a 100 ms duration. This was repeated for 100 training epochs, when the receptive field converged. By observing the final synaptic weights connecting to each neuron for the patterns that were presented to the network, the output neuron selective to each of the 10 patterns was identified. The spiking rates of each neuron was monitored and recorded into a raster plot, clearly showing the spiking activity per pattern. This was used to determine if each neuron classified a single pattern, and did not dominate the output response.

4.6 Single-Layer Network Results

In Fig. 6, we depict the excitatory synapse receptive field pre- and post-learning. Fig. 6(a) shows the initial randomised memristive weights, whilst Fig. 6(b) shows the final weights after 100 training epochs. As shown, each of the ten neurons learns to converge to a particular pattern. To visualise the learning process of the network, Fig. 7 was generated. This figure shows the spiking raster plots of all the 10 output neurons. Fig. 7(c) confirms that, at the end of the learning process, each neuron only conforms to one pattern, as each neuron mostly fires for one 100ms period, barring minor spiking events for neurons 7, 8 and 10. On occasions, it was rarely observed that neurons that learned patterns “8” and “6” would have some overlaps in the raster plot. The reason for this is that the patterns are similar and have many overlapping pixels. Furthermore, patterns “0”, ”3” and “5” share some major similarities with “8” and “6” as well, and would also overlap on rare occasions. The results shown in Fig. 6 and Fig. 7 demonstrate that our proposed synaptic memristive device is able to effectively perform unsupervised learning.

5 Handwritten Digit Classification

After confirming that our graphene-based memristive synapse can realise STDP and perform unsupervised pattern classification, we investigated its ability to carry out a more complex classification task. In most of the literature, one of the common benchmarks used to evaluate the performance of neuromorphic systems is the classification of handwritten digits using the MNIST dataset. To truly compare our graphene memristive device to related works in literature, it was decided that another larger network had to be created in order to test our device’s synaptic performance in a SNN. This network was similar to our single-layer network created previously, as the neuron model and

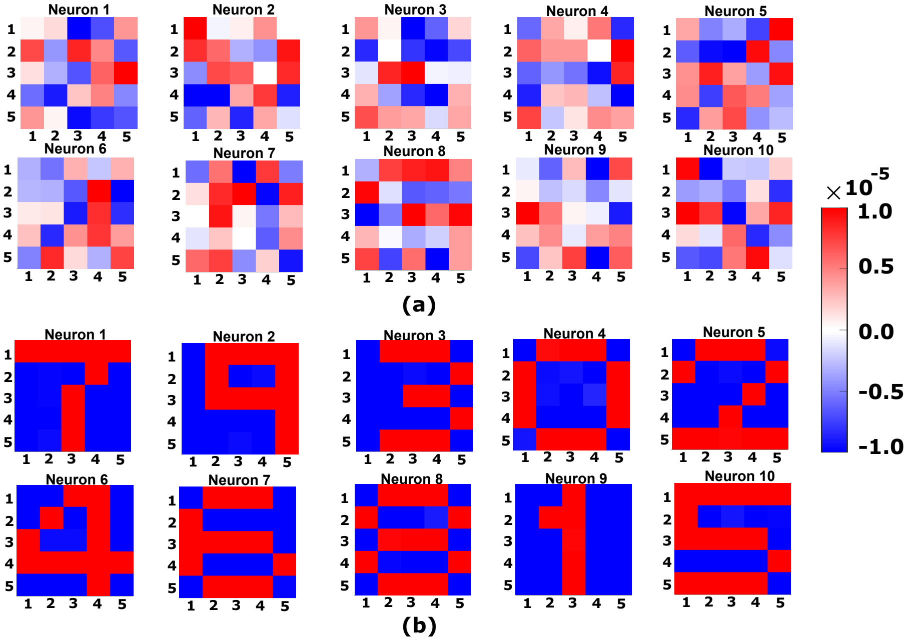


Figure 6: (a) Initial randomised memristive synaptic weights for the single-layer network. (b) Memristive synaptic weights after 100 training epochs.

STDP window used were kept the same. However, due to the larger number of pixels of the MNIST images, we increased the number of output spiking neurons in order to ensure unsupervised learning occurred.

5.1 Neuron Parameter Modification

The first modification included some parameter adjustments of the LIF neuron model used previously. This was to accommodate for the larger sum of current being presented to the network. The parameter changes are summarised in Table 2.

5.2 Input Encoding

The training and test input images were derived from the MNIST dataset, which consists of 60,000 training examples and 10,000 test examples. Each image (example) consists of 28×28 pixels encoded in grayscale between 0 and 255. We investigated different ways to encode the input image grayscale pixel values. These included the mapping of the pixel value to the average frequency of a Poissonian spike train, as well as using a binary mapping scheme. In this scheme, if the pixel value is above/below a predetermined threshold, a Poissonian spike train with a high/low average frequency, represents that pixel. Our experimentation showed that the binary encoding works well and could result in a bi-modal STDP weight convergence during training, similar to the

Table 2: Summary of neuron parameters used in the binary and MNIST classification tasks.

Parameter	Binary	MNIST
τ	0.0015 s	3 s
R	500 Ω	1000 Ω
E_{Rest}	0.0095 V	0 V
$\mathrm{V}_{th_{min}}$	0.010 V	0.25 V
$\mathrm{V}_{th_{max}}$	0.030 V	30 V
T	16 Post-Synaptic Spikes	1 Post-Synaptic Spike
γ	0.029 V/s	1×10^{-6}
$\mathrm{I}_{Inhibitory}$	-6$\times 10^{-5}$A	-6.02 $\times 10^{-3}$A

outcome seen in Fig. 6(b). Consequently, through experimentation, we chose a threshold value of 50 to encode all MNIST images to a series of Poissonian spike trains.

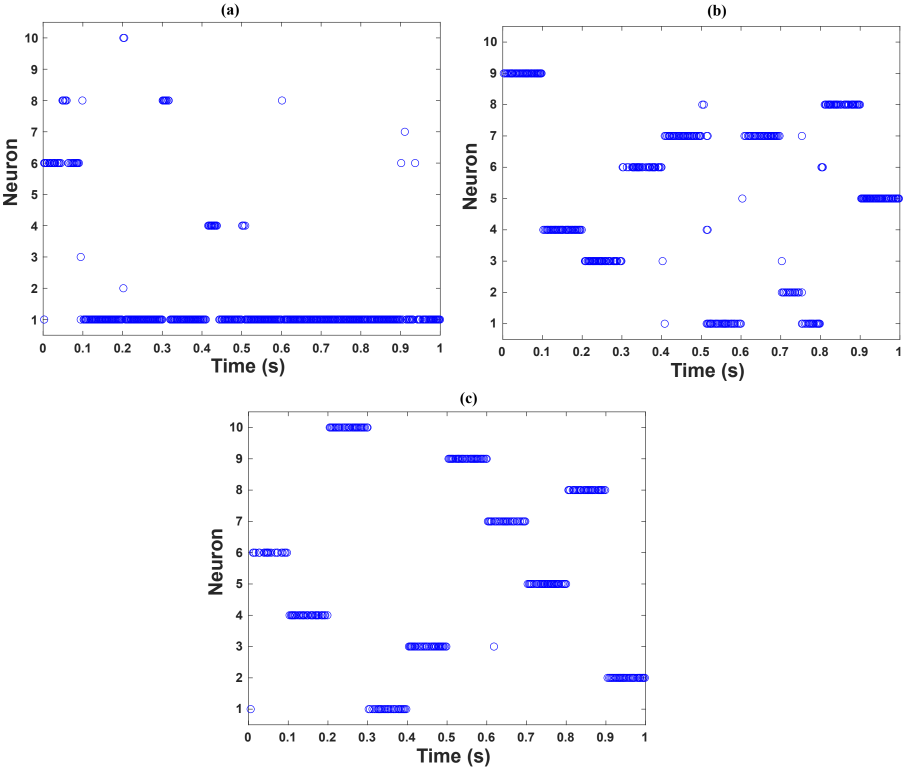


Figure 7: Various raster plots of single layer network. (a) Raster plot during initial epoch. (b) Raster plot during the 10th epoch. (c) Raster plot during the last epoch.

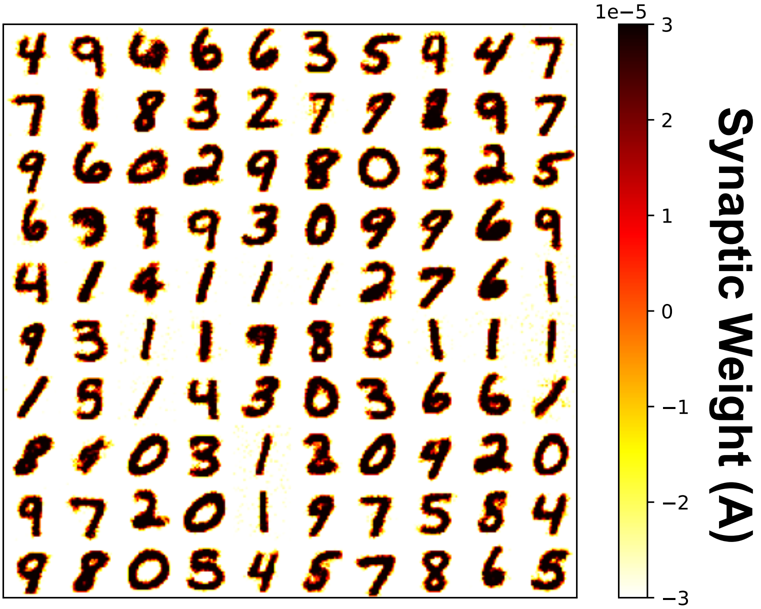


Figure 8: Receptive field after training the MNIST classification network.

5.3 Training and Classification

The training phase adopted was similar to the one used to train our single-layer network. However, each image was presented for 50 ms (as opposed to 100 ms) to allow for weight and threshold updates. Images were also presented in a random order to eliminate any form of bias, and were only selected from the MNIST training set, i.e., no image from the MNIST test set was used to train the network. In order to label the network output neurons, each neuron was assigned a class. This assigning procedure was done by monitoring each neuron's spiking activity in response to every class presented. For a given

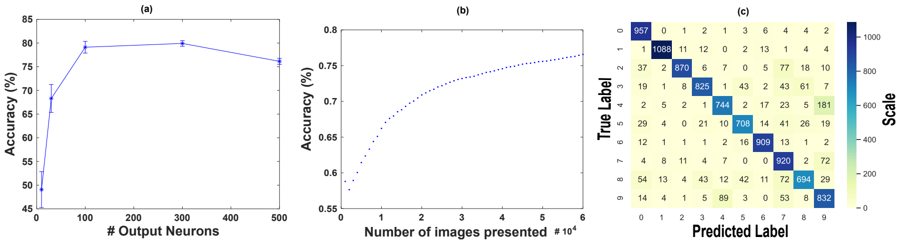


Figure 9: (a) Classification accuracy plotted against the number of output neurons for a single training epoch. (b) Running accuracy during the training phase for a network with 100 output neurons. (c) Confusion matrix after the evaluation phase of MNIST classification for a network with 100 output neurons.

neuron, the class with the highest spiking activity during the training phase was assigned to that neuron.

To evaluate the network after training, we used a similar method to that used in [32], which proposed a classification accuracy where the output neuron with the highest spiking activity was said to classify the input pattern being presented. If the class assigned to the neuron matches the class presented to the network, then the pattern was deemed to be successfully classified. Thus the accuracy could be measured by dividing the total number of correctly classified digits with the total number of digits presented.

5.4 Parameter Optimisation

Prior to training and evaluating our MNIST classification network architecture with various number of output neurons, we performed preliminary investigations to determine optimal network hyper-parameters for each of these networks. After an initial exploratory analysis, empirically, it was found that the γ and τ parameters had the largest influence on network performance. Consequently, Bayesian Optimisation was used to optimise γ and τ to maximise the classification accuracy. For all network configurations, other neuron parameters were kept fixed, as shown in Table 2. To confine the search space, when performing Bayesian Optimisation, γ and τ were confined between 0.05–0.15, and 0.0001–0.01, respectively.

To perform hyper-parameter optimisation, separate training and validation subsets were constructed from the original MNIST training set using `sklearn.model_selection.train_test_split` with an initial random state of 1, and with shuffling enabled. Both subsets adhered to the same class distribution as the MNIST test set, and were sampled from the pseudo-randomly shuffled original MNIST training set. The training subset contained 50,000 images, and the validation subset contained 10,000 images. A total of 250 Bayesian trials were executed for each network configuration, sampling γ and τ parameters in log-space. Trial pruning was used to predicatively prune unsuccessful trials to speed up optimisation using the `optuna` [79] Python library.

After performing hyper-parameter optimisation, the classification accuracy was reported for each network configuration by training each network using the original MNIST training set, and evaluating each network using the MNIST test set. In Table 3, we report the best validation accuracy achieved and the optimised network parameters for network configurations with different numbers of output neurons. We believe that the validation accuracies achieved were generally higher than the test set accuracies due to the inability of our constructed validation set to generalise, i.e., to be completely representative of, to the MNIST test set. While cross validation could be used to improve the efficacy of our hyper-parameter optimisation procedure, it is computationally demanding, and would exponentially increase required run-times. Consequently, we leave further hyper-parameter optimisation investigations to future work. Note that, except for the 10 and 30 output neuron cases, which

Table 3: Optimised network parameters and the best validation accuracy achieved in 250 optimisation trials.

N. Output Neurons	**Validation Accuracy (%)**	γ	τ
10	44.48	4.80	2.66×10^{-5}
30	66.46	2.69	$3.25\ \times10^{-8}$
100	80.87	2.84	5.95×10^{-8}
300	85.66	2.56	2.18×10^{-8}
500	85.64	2.29	1.80×10^{-8}

use the parameters in Table 2, the results reported in Section 5.5 were obtained using the parameters that achieved the best results when classifying the MNIST test set.

5.5 Results

After performing parameter optimisation, the classification accuracy was investigated for each network configuration by training the network using the original 60,000 images in the MNIST training set, and testing it using the 10,000 images in the MNIST test set. In addition to the optimised γ and τ parameter sets reported in Table 3, the classification accuracy was determined for other γ and τ parameter sets, which achieved significant performance on the validation set. In this section, we report our best results, i.e., those that obtained the highest classification accuracy on the MNIST test set, as opposed to the validation set constructed during the optimisations. As such, not all of the parameters from Table 3 were used, as we believe our constructed validation set failed to generalise to (i.e. to be completely representative of) the MNIST test set. It is believed that testing on the validation set as opposed to the actual test set contributed to these reductions in performance. Furthermore, it is also possible that the randomised variations between each training and testing evaluation is another likely contributor to this effect. All parameters used to train networks whose evaluation results are presented in this section are provided[1].

Fig. 8 illustrates the synaptic receptive field after a sample training phase of the network. The receptive field clearly shows learning of the handwritten digits. This is reflected in the high average test-set classification accuracies shown in Fig. 9(a), where the values are calculated over 10 trials for networks with different number of output neurons. For each trial, the network was trained from scratch starting with random initial weights to learn the MNIST training set, and after training was completed, it was tested using the MNIST test set. The figure also shows the standard deviation across these 10 trials. It is worth noting that, these classification accuracies are reported for a single training epoch, which shows the efficiency of our network in reaching a great performance while being trained only for one epoch. As shown, an increase in the number of neurons has resulted in more neurons learning more variations of

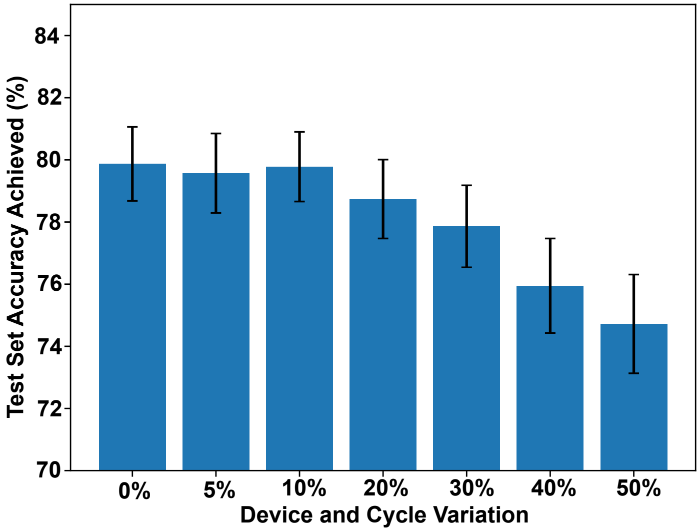


Figure 10: The effect of device and cycle variations on MNIST test set classification accuracy, for when 100 output neurons were trained for one epoch.

each pattern. This gives a wider representation of all of the possible handwritten digits, resulting in a higher accuracy. It was also found that the standard deviation of the results obtained generally decreased with the increase in the number of neurons except for the 500 case. It is believed that one epoch may be too short to train all 500 neurons accurately, thus leading to this result.

Fig. 9(b) demonstrates the classification accuracy plotted against the number of images presented to the network in a training epoch. Naturally, it was expected that increasing the number of training images increased the accuracy, due to the more thorough learning. The leveling off of the accuracy was also expected, and is in keeping with other networks in literature that have attempted this [31].

Finally, analysing the confusion matrix in Fig. 9(c) shows which digits were more accurately classified than others. The most accurately classified digits were "0" and "1", whilst the most inaccurately classified digits were "3","4" and "9". The patterns "4" and "9" were often misclassified with each other. This observation is in agreement with literature [31]. In addition, the pattern "3" was often misclassified with "5" and "8", although in much lesser amounts than "4" and "9". This result has also been observed in other literature such as [36], further re-enforcing the results that have been obtained.

As the learning was unsupervised, the number of output neurons being assigned to a class could not be controlled. One common observation in the results was an over-representation of the class "1" being present in the final receptive field. Naturally, this meant that this pattern was the most accurately classified pattern, as shown in the confusion matrix in Fig. 9(c). It was also noted that the pattern "0" was consistently under-represented, although its accuracy remained relatively high. A possible explanation could be that within the approximately 6,000 MNIST training images of the pattern "1", there exists a wider variety of ways in which to represent "1" than any other character. Thus, more output neurons converge to this pattern to better reflect this.

In these simulations, we also tested our network's stability when device-to-device and cycle-to-cycle variations are considered. Although insufficient device data was available to perform device specific modelling, it is still possible to consider the effects of device variations by sampling values of device current through normal distributions with different widths similar to [80]. Modelling the device-to-device variations was done by sampling each device current value from a normal distribution around the averaged device current shown in Fig. 10, where the standard deviation of the normal distribution was varied from 0% to 50%. This is equivalent to having a different STDP window for each of the network devices. Cycle-to-cycle variations were modelled by further sampling each value from the STDP window (which is now device specific due to the device-to-device variation) from a normal distribution. The results shown in Fig. 10 clearly illustrate the drop in performance when the variations are increased. Note that the same standard deviation values for both cycle-to-cycle and device-to-device variations were used.

5.6 Comparison

Fig. 11 shows the comparison between our best results and state-of-the-art literature on MNIST classification using unsupervised learning in SNNs. Our results show the average test set classification accuracy of 10 learning trials. As shown, when a low number of output neurons are used, our network shows a low accuracy with a high variation reflected in its high standard deviation. This can be attributed to our input encoding scheme, as well as the homeostatic mechanism we have employed. In addition, we have only used one training epoch. Note that, [33] has used a similar homeostatic neuronal spiking regulation, which resulted in a slightly better accuracy for 10 output neurons. This slight improvement is most likely due to its more complex input encoding scheme, and higher number of training epoch.

Furthermore, most of the STDP-based MNIST classification studies in literature [34–38] have used up to 100 neurons, which has resulted in classification accuracies lower than 80%, which is not significant. Previous studies [31–33] have shown that this problem can be addressed by including more output neurons to improve the classification accuracy.

Compared to state-of-the-art, our network demonstrates on-par performance for the case where 100 output neurons. In addition, it is better than or comparable with other works that reported accuracies for 300 and 500 neuron cases. Although the 300 neuron case did not demonstrate the above 90% accuracy reported by [33], the network still performed quite well for the cases where a larger number of output neurons were used and was on par with the software-based work of Diehl and Cook [31], while showing a better performance than [32] for 500 neurons.

The lower classification accuracy compared to [33] can be due to our simple input image binary encoding scheme, which may result in losing some information contained within the images. The benefit, however, is a much simpler encoding scenario, which requires less complex spike encoding circuitry. When the 100, 300 and 500 neuron cases were trained for only 1 epoch (shown in Fig. 9(a)), the network achieved comparable accuracies to most of the works in literature that were trained for more epochs as shown in Table 1. This suggests a very swift convergence rate, and that the network is being trained in an efficient manner.

It is important to note that whilst these results show decent performance for MNIST classification, there are several limitations to this study. One limitation is the device itself, with its extra transitions close to the origin. These additional transitions means the device is volatile, and can only operate with a memory of approximately 500 ms. This limits the synapse implementation and needs additional memory circuits for long term storage. Furthermore, the device shows a high degree of variability, which can result in inefficient learning. However, this variability has been addressed in previous literature in many ways [81–83], and these methods could be adapted, but this comes at the expense of increased complexity.

6 Conclusion

In this paper, we investigated the potential of graphene-based memristive devices in neuromorphic SNNs for unsupervised learning. We demonstrated that the device's I-V characteristics can be used to mimic biology in producing the excitatory post synaptic current response to the differential voltage applied to the memristive synapse, as pre- and post-synaptic potentials. Using this, we developed an STDP window and performed two different pattern classification tasks. We showed that a homeostatic neuronal activity regulation could help the STDP-based unsupervised learning to perform MNIST classification efficiently, even when a simple binary pattern encoding scheme is used. Our results demonstrated improvement in classification accuracies compared to other memristive SNNs in literature when a higher number of output neurons are used to reach significant classification performance. We believe that this work will serve as a foundation and benchmark for future neuromorphic architectures utilising graphene memristive devices. However, further investigations could be provided into the challenges of utilising graphene in memristive devices, such as solving the issue of the fairly large current draw in the on-state using current limiters/selectors, or investigating potential second order mechanisms to model graphene-based devices.

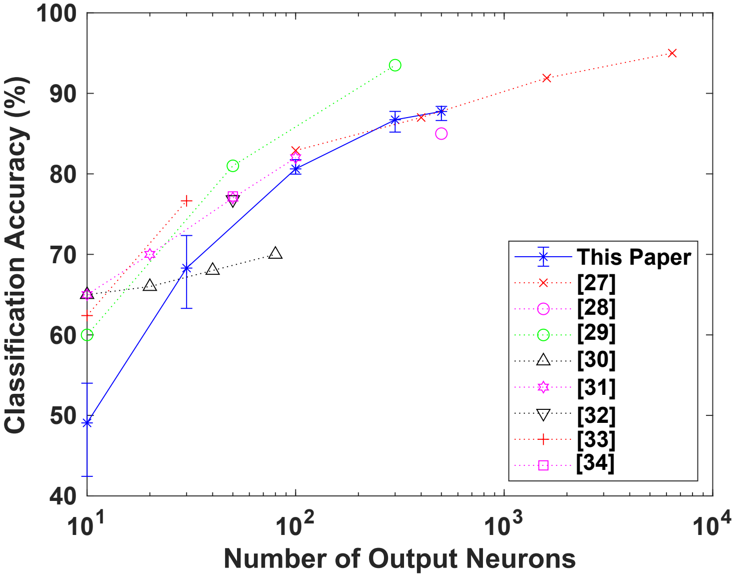


Figure 11: Comparison of SNNs that have performed unsupervised MNIST classification using memristive synapses. [31] is the only software-based network reported in this figure. For our work, one training epoch was used for when the number of output neurons was equal to 10 and 30, 2 training epochs were used when the number of output neurons was equal to 100, and 3 epochs were used when the number of output neurons was equal to either 300 or 500.

Acknowledgments. Ben Walters acknowledges the Domestic Research Training Program Scholarship (DRTPS) (Australia). Corey Lammie acknowledges the Domestic Prestige Research Training Program Scholarship (DPRTPS) (Australia), IBM PhD Fellowship (IBM, US), and the Circuits and Systems (CAS) Society Predoctoral Grant (IEEE, US).

Declarations

- Funding: The authors have no relevant financial or non-financial interests to disclose
- Competing interests: The authors have no competing interests to declare that are relevant to the content of this article.
- Ethics approval: Not Applicable
- Consent to participate: Not Applicable
- Consent for publication: All authors
- Availability of data and materials: The data and materials will be made publicly available

- Code availability: Upon acceptance our code will be made publicly available. For the review process is it provided at https://anonymous.4open.science/r/SNN-GrapheneSynapses-2E0D

References

[1] Chua L. Memristor-The missing circuit element [Journal Article]. IEEE Transactions on Circuit Theory. 1971;18(5):507–519. https://doi.org/10.1109/TCT.1971.1083337.

[2] Maranhão G, Guimarães JG. Low-power hybrid memristor-CMOS spiking neuromorphic STDP learning system. IET Circuits, Devices & Systems. 2021;15(3):237–250. https://doi.org/https://doi.org/10.1049/cds2.12018. https://arxiv.org/abs/https://ietresearch.onlinelibrary.wiley.com/doi/pdf/10.1049/cds2.12018.

[3] Rahimi Azghadi M, Chen YC, Eshraghian JK, Chen J, Lin CY, Amirsoleimani A, et al. Complementary Metal-Oxide Semiconductor and Memristive Hardware for Neuromorphic Computing. Advanced Intelligent Systems. 2020;2(5):1900189. https://doi.org/https://doi.org/10.1002/aisy.201900189. https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/aisy.201900189.

[4] Strukov DB, Snider GS, Stewart DR, Williams RS. The missing memristor found. Nature. 2008;453(7191):80–83. https://doi.org/10.1038/nature06932.

[5] Sun K, Chen J, Yan X. The Future of Memristors: Materials Engineering and Neural Networks. Advanced Functional Materials. 2021;31(8):2006773. https://doi.org/https://doi.org/10.1002/adfm.202006773. https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/adfm.202006773.

[6] Ambrogio S, Ciocchini N, Laudato M, Milo V, Pirovano A, Fantini P, et al. Unsupervised Learning by Spike Timing Dependent Plasticity in Phase Change Memory (PCM) Synapses. Frontiers in Neuroscience. 2016;10:56. https://doi.org/10.3389/fnins.2016.00056.

[7] Velev JP, Duan CG, Burton JD, Smogunov A, Niranjan MK, Tosatti E, et al. Magnetic Tunnel Junctions with Ferroelectric Barriers: Prediction of Four Resistance States from First Principles. Nano Lett. 2009 Jan;9(1):427–432. https://doi.org/10.1021/nl803318d.

[8] Garcia V, Fusil S, Bouzehouane K, Enouz-Vedrenne S, Mathur ND, Barthélémy A, et al. Giant tunnel electroresistance for non-destructive

readout of ferroelectric states. Nature. 2009;460(7251):81–84. https://doi.org/10.1038/nature08128.

[9] Solanki A, Guerrero A, Zhang Q, Bisquert J, Sum TC. Interfacial Mechanism for Efficient Resistive Switching in Ruddlesden-Popper Perovskites for Non-volatile Memories. J Phys Chem Lett. 2020 Jan;11(2):463–470. https://doi.org/10.1021/acs.jpclett.9b03181.

[10] Ahn M, Park Y, Lee SH, Chae S, Lee J, Heron JT, et al. Memristors Based on (Zr, Hf, Nb, Ta, Mo, W) High-Entropy Oxides. Advanced Electronic Materials. 2021;7(5):2001258. https://doi.org/https://doi.org/10.1002/aelm.202001258. https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/aelm.202001258.

[11] Zhang C, Zhou H, Chen S, Zhang G, Yu ZG, Chi D, et al. Recent progress on 2D materials-based artificial synapses. Critical Reviews in Solid State and Materials Sciences. 2021;0(0):1–26. https://doi.org/10.1080/10408436.2021.1935212. https://arxiv.org/abs/https://doi.org/10.1080/10408436.2021.1935212.

[12] Kwon KC, Baek JH, Hong K, Kim SY, Jang HW. Memristive Devices Based on Two-Dimensional Transition Metal Chalcogenides for Neuromorphic Computing. Nano-Micro Letters. 2022;14(1):58. https://doi.org/10.1007/s40820-021-00784-3.

[13] Zhou Z, Yang F, Wang S, Wang L, Wang X, Wang C, et al. Emerging of two-dimensional materials in novel memristor. Frontiers of Physics. 2021;17(2):23204. https://doi.org/10.1007/s11467-021-1114-5.

[14] Jacob MV, Rawat RS, Ouyang B, Bazaka K, Kumar DS, Taguchi D, et al. Catalyst-Free Plasma Enhanced Growth of Graphene from Sustainable Sources. Nano Letters. 2015;15(9):5702–5708. PMID: 26263025. https://doi.org/10.1021/acs.nanolett.5b01363. https://arxiv.org/abs/https://doi.org/10.1021/acs.nanolett.5b01363.

[15] Burr GW. A role for analogue memory in AI hardware. Nature Machine Intelligence. 2019;1(1):10–11. https://doi.org/10.1038/s42256-018-0007-y.

[16] Wang J, Zhuge F. Memristive Synapses for Brain-Inspired Computing. Advanced Materials Technologies. 2019;4(3):1800544. https://doi.org/https://doi.org/10.1002/admt.201800544. https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/admt.201800544.

[17] Xiao Z, Yan B, Zhang T, Huang R, Yang Y. Memristive devices based hardware for unlabeled data processing. Neuromorphic Computing and Engineering. 2022 jun;2(2):022003. https://doi.org/10.1088/2634-4386/

ac734a.

[18] Huang M, Li Z, Zhu H. Recent Advances of Graphene and Related Materials in Artificial Intelligence. Advanced Intelligent Systems. 2022;4(10):2200077.

[19] Li R, Huang P, Feng Y, Zhou Z, Zhang Y, Ding X, et al. Hardware demonstration of srdp neuromorphic computing with online unsupervised learning based on memristor synapses. Micromachines. 2022;13(3):433.

[20] Xiang Y, Huang P, Zhao Y, Zhao M, Gao B, Wu H, et al. Impacts of State Instability and Retention Failure of Filamentary Analog RRAM on the Performance of Deep Neural Network. IEEE Transactions on Electron Devices. 2019 Nov;66(11):4517–4522. https://doi.org/10.1109/TED.2019.2931135.

[21] Vaila R, Chiasson J, Saxena V. A Deep Unsupervised Feature Learning Spiking Neural Network With Binarized Classification Layers for the EMNIST Classification. IEEE Transactions on Emerging Topics in Computational Intelligence. 2020 Feb;6(1):124–135. https://doi.org/10.1109/TETCI.2020.3035164.

[22] Chellappa R, Theodoridis S, van Schaik A. Advances in Machine Learning and Deep Neural Networks. Proceedings of the IEEE. 2021 May;109(5):607–611. https://doi.org/10.1109/JPROC.2021.3072172.

[23] Zhang W, Li P. Information-Theoretic Intrinsic Plasticity for Online Unsupervised Learning in Spiking Neural Networks. Frontiers in Neuroscience. 2019;13:31. https://doi.org/10.3389/fnins.2019.00031.

[24] Zambrano D, Nusselder R, Scholte HS, Bohté SM. Sparse Computation in Adaptive Spiking Neural Networks. Frontiers in Neuroscience. 2019;12:987. https://doi.org/10.3389/fnins.2018.00987.

[25] Zenke F, Ganguli S. SuperSpike: Supervised Learning in Multilayer Spiking Neural Networks. Neural Computation. 2017 05;30. https://doi.org/10.1162/neco_a_01086.

[26] Neftci EO, Mostafa H, Zenke F. Surrogate Gradient Learning in Spiking Neural Networks: Bringing the Power of Gradient-Based Optimization to Spiking Neural Networks. IEEE Signal Processing Magazine. 2019 Nov;36(6):51–63. https://doi.org/10.1109/MSP.2019.2931595.

[27] Eshraghian JK, Ward M, Neftci E, Wang X, Lenz G, Dwivedi G, et al. Training spiking neural networks using lessons from deep learning. arXiv preprint arXiv:210912894. 2021;.

[28] Azghadi MR, Linares-Barranco B, Abbott D, Leong PH. A Hybrid CMOS-Memristor Neuromorphic Synapse [Journal Article]. IEEE Trans Biomed Circuits Syst. 2017;11(2):434–445. https://doi.org/10.1109/TBCAS.2016.2618351.

[29] Rahimi Azghadi M, Al-Sarawi S, Abbott D, Iannella N. A neuromorphic VLSI design for spike timing and rate based synaptic plasticity. Neural Networks. 2013;45:70–82. Neuromorphic Engineering: From Neural Systems to Brain-Like Engineered Systems. https://doi.org/https://doi.org/10.1016/j.neunet.2013.03.003.

[30] Bi Gq, Poo Mm. Synaptic Modifications in Cultured Hippocampal Neurons: Dependence on Spike Timing, Synaptic Strength, and Postsynaptic Cell Type. Journal of Neuroscience. 1998;18(24):10464–10472. https://doi.org/10.1523/JNEUROSCI.18-24-10464.1998. https://arxiv.org/abs/https://www.jneurosci.org/content/18/24/10464.full.pdf.

[31] Diehl P, Cook M. Unsupervised learning of digit recognition using spike-timing-dependent plasticity. Frontiers in Computational Neuroscience. 2015;9:99. https://doi.org/10.3389/fncom.2015.00099.

[32] Brivio S, Ly DRB, Vianello E, Spiga S. Non-linear Memristive Synaptic Dynamics for Efficient Unsupervised Learning in Spiking Neural Networks. Frontiers in Neuroscience. 2021;15:27. https://doi.org/10.3389/fnins.2021.580909.

[33] Querlioz D, Bichler O, Dollfus P, Gamrat C. Immunity to Device Variations in a Spiking Neural Network With Memristive Nanodevices. IEEE Transactions on Nanotechnology. 2013 05;12:288–295. https://doi.org/10.1109/TNANO.2013.2250995.

[34] Kim S, Yoon J, Kim HD, Choi SJ. Carbon Nanotube Synaptic Transistor Network for Pattern Recognition. ACS Appl Mater Interfaces. 2015 Nov;7(45):25479–25486. https://doi.org/10.1021/acsami.5b08541.

[35] Hansen M, Zahari F, Ziegler M, Kohlstedt H. Double-Barrier Memristive Devices for Unsupervised Learning and Pattern Recognition. Frontiers in Neuroscience. 2017;11:91. https://doi.org/10.3389/fnins.2017.00091.

[36] Guo Y, Wu H, Gao B, Qian H. Unsupervised Learning on Resistive Memory Array Based Spiking Neural Networks. Frontiers in Neuroscience. 2019;13:812. https://doi.org/10.3389/fnins.2019.00812.

[37] Nandakumar SR, Rajendran B. Bio-mimetic Synaptic Plasticity and Learning in a sub-500mV Cu/SiO2/W Memristor [Journal Article]. Microelectronic Engineering. 2020;226. https://doi.org/10.1016/j.mee.2020.111290.

[38] Boybat I, Le Gallo M, Nandakumar SR, Moraitis T, Parnell T, Tuma T, et al. Neuromorphic computing with multi-memristive synapses. Nature Communications. 2018;9(1):2514. https://doi.org/10.1038/s41467-018-04933-y.

[39] Bill J, Legenstein R. A compound memristive synapse model for statistical learning through STDP in spiking neural networks. Frontiers in Neuroscience. 2014;8. https://doi.org/10.3389/fnins.2014.00412.

[40] Covi E, Brivio S, Serb A, Prodromakis T, Fanciulli M, Spiga S. Analog Memristive Synapse in Spiking Networks Implementing Unsupervised Learning [Journal Article]. Front Neurosci. 2016;10:482. https://doi.org/10.3389/fnins.2016.00482.

[41] Qu L, Zhao Z, Wang L, Wang Y. Efficient and hardware-friendly methods to implement competitive learning for spiking neural networks. Neural Computing and Applications. 2020;32(17):13479–13490 . https://doi.org/10.1007/s00521-020-04755-4.

[42] Demin VA, Nekhaev DV, Surazhevsky IA, Nikiruy KE, Emelyanov AV, Nikolaev SN, et al. Necessary conditions for STDP-based pattern recognition learning in a memristive spiking neural network. Neural Networks. 2021;134:64–75. https://doi.org/https://doi.org/10.1016/j.neunet.2020.11.005.

[43] Hajiabadi Z, Shalchian M. Memristor-based synaptic plasticity and unsupervised learning of spiking neural networks. Journal of Computational Electronics. 2021;20(4):1625–1636 . https://doi.org/10.1007/s10825-021-01719-2.

[44] Kopelevich Y, Bud'ko S, Cooper DR, D'Anjou B, Ghattamaneni N, Harack B, et al. Experimental Review of Graphene. ISRN Condensed Matter Physics. 2012;2012:501686. https://doi.org/10.5402/2012/501686.

[45] Akinwande D, Brennan CJ, Bunch JS, Egberts P, Felts JR, Gao H, et al. A review on mechanics and mechanical properties of 2D materials—Graphene and beyond. Extreme Mechanics Letters. 2017;13:42–77. https://doi.org/https://doi.org/10.1016/j.eml.2017.01.008.

[46] Chen Y, Zhou Y, Zhuge F, Tian B, Yan M, Li Y, et al. Graphene-ferroelectric transistors as complementary synapses for supervised learning in spiking neural network. npj 2D Materials and Applications. 2019;3(1):31. https://doi.org/10.1038/s41699-019-0114-6.

[47] Schranghamer TF, Oberoi A, Das S. Graphene memristive synapses for high precision neuromorphic computing. Nature Communications. 2020;11(1):5474. https://doi.org/10.1038/s41467-020-19203-z.

[48] Sun Y, Lin Y, Zubair A, Xie D, Palacios T. WSe2/graphene heterojunction synaptic phototransistor with both electrically and optically tunable plasticity. 2D Materials. 2021 may;8(3):035034. https://doi.org/10.1088/2053-1583/abfa6a.

[49] Abunahla H, Halawani Y, Alazzam A, Mohammad B. NeuroMem: Analog Graphene-Based Resistive Memory for Artificial Neural Networks. Scientific Reports. 2020;10(1):9473. https://doi.org/10.1038/s41598-020-66413-y.

[50] Qi M, Cao S, Yang L, You Q, Shi L, Wu Z. Uniform multilevel switching of graphene oxide-based RRAM achieved by embedding with gold nanoparticles for image pattern recognition. Applied Physics Letters. 2020;116(16):163503. https://doi.org/10.1063/5.0003696. https://arxiv.org/abs/https://doi.org/10.1063/5.0003696.

[51] Romero FJ, Toral A, Medina-Rull A, Moraila-Martinez CL, Morales DP, Ohata A, et al. Resistive Switching in Graphene Oxide. Frontiers in Materials. 2020;7:17. https://doi.org/10.3389/fmats.2020.00017.

[52] Porro S, Accornero E, Pirri CF, Ricciardi C. Memristive devices based on graphene oxide. Carbon. 2015;85:383–396. https://doi.org/https://doi.org/10.1016/j.carbon.2015.01.011.

[53] Liu B, Liu Z, Chiu IS, Di M, Wu Y, Wang JC, et al. Programmable Synaptic Metaplasticity and below Femtojoule Spiking Energy Realized in Graphene-Based Neuromorphic Memristor. ACS Appl Mater Interfaces. 2018 Jun;10(24):20237–20243. https://doi.org/10.1021/acsami.8b04685.

[54] Krishnaprasad A, Choudhary N, Das S, Dev D, Kalita H, Chung HS, et al. Electronic synapses with near-linear weight update using MoS2/graphene memristors. Applied Physics Letters. 2019;115(10):103104. https://doi.org/10.1063/1.5108899. https://arxiv.org/abs/https://doi.org/10.1063/1.5108899.

[55] Feng X, Liu X, Ang KW. 2D photonic memristor beyond graphene: progress and prospects [Journal Article]. Nanophotonics. 2020;0(0). https://doi.org/10.1515/nanoph-2019-0543.

[56] Cao G, Meng P, Chen J, Liu H, Bian R, Zhu C, et al. 2D Material Based Synaptic Devices for Neuromorphic Computing. Advanced Functional Materials. 2021;31(4):2005443. https://doi.org/https://doi.org/10.1002/adfm.202005443. https://arxiv.org/abs/https://onlinelibrary.wiley.com/doi/pdf/10.1002/adfm.202005443.

[57] Yalagala B, Khandelwal S, J D, Badhulika S. Wirelessly destructible MgO-PVP-Graphene composite based flexible transient memristor for

security applications. Materials Science in Semiconductor Processing. 2019;104:104673. https://doi.org/https://doi.org/10.1016/j.mssp.2019.104673.

[58] Bi G, Poo M. Synaptic modification by correlated activity: Hebb's postulate revisited [Journal Article]. Annu Rev Neurosci. 2001;24:139–66. https://doi.org/10.1146/annurev.neuro.24.1.139.

[59] Sahu DP, Jetty P, Jammalamadaka SN. Graphene oxide based synaptic memristor device for neuromorphic computing. Nanotechnology. 2021 jan;32(15):155701. https://doi.org/10.1088/1361-6528/abd978.

[60] Wang H, Laurenciu NC, Jiang Y, Cotofana SD. Compact Graphene-Based Spiking Neural Network With Unsupervised Learning Capabilities. IEEE Open Journal of Nanotechnology. 2020;1:135–144. https://doi.org/10.1109/OJNANO.2020.3041198.

[61] Wang H, Laurenciu NC, Jiang Y, Cotofana S. Graphene-Based Artificial Synapses with Tunable Plasticity. J Emerg Technol Comput Syst. 2021 Jun;17(4). https://doi.org/10.1145/3447778.

[62] Wang Z, Liu C, Deng Y, Huang Z, He S, Guo D. Carbon-based Spiking Neural Network Implemented with Single-Electron Transistor and Memristor for Visual Perception. In: 2020 IEEE 14th International Conference on Anti-counterfeiting, Security, and Identification (ASID); 2020. p. 143–146.

[63] Wang H, Cucu Laurenciu N, Cotofana S. A Reconfigurable Graphene-based Spiking Neural Network Architecture. IEEE Open Journal of Nanotechnology. 2021;p. 1–1. https://doi.org/10.1109/OJNANO.2021.3094761.

[64] Hajri B, Aziza H, Mansour MM, Chehab A. RRAM Device Models: A Comparative Analysis With Experimental Validation. IEEE Access. 2019;7:168963–168980. https://doi.org/10.1109/ACCESS.2019.2954753.

[65] Pershin Y, Di Ventra M. SPICE Model of Memristive Devices with Threshold. Radioengineering. 2012 04;22.

[66] Kvatinsky S, Ramadan M, Friedman EG, Kolodny A. VTEAM: A General Model for Voltage-Controlled Memristors [Journal Article]. IEEE Transactions on Circuits and Systems II: Express Briefs. 2015;62(8):786–790. https://doi.org/10.1109/tcsii.2015.2433536.

[67] Messaris I, Serb A, Stathopoulos S, Khiat A, Nikolaidis S, Prodromakis T. A Data-Driven Verilog-A ReRAM Model. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. 2018

Dec;37(12):3151–3162. https://doi.org/10.1109/TCAD.2018.2791468.

[68] Jacob MV, Taguchi D, Iwamoto M, Bazaka K, Rawat RS. Resistive switching in graphene-organic device: Charge transport properties of graphene-organic device through electric field induced optical second harmonic generation and charge modulation spectroscopy [Journal Article]. Carbon. 2017;112:111–116. https://doi.org/10.1016/j.carbon.2016.11.005.

[69] Rahimi Azghadi M, Moradi S, Fasnacht D, Ozdas M, Indiveri G. Programmable Spike-Timing-Dependent Plasticity Learning Circuits in Neuromorphic VLSI Architectures [Journal Article]. ACM Journal on Emerging Technologies in Computing Systems. 2015;12(2). https://doi.org/10.1145/2658998.

[70] Izhikevich EM. Simple model of spiking neurons. IEEE Transactions on Neural Networks. 2003 Nov;14(6):1569–1572. https://doi.org/10.1109/TNN.2003.820440.

[71] Hodgkin AL, Huxley AF. A quantitative description of membrane current and its application to conduction and excitation in nerve. Bulletin of Mathematical Biology. 1990;52(1):25–71. https://doi.org/10.1007/BF02459568.

[72] Lapique L. Recherches quantitatives sur l'excitation electrique des nerfs traitee comme une polarization [Journal Article]. Journal of Physiology and Pathololgy. 1907;9:620–635.

[73] Zamarreno-Ramos C, Camunas-Mesa LA, Perez-Carrasco JA, Masquelier T, Serrano-Gotarredona T, Linares-Barranco B. On spike-timing-dependent-plasticity, memristive devices, and building a self-learning visual cortex [Journal Article]. Front Neurosci. 2011;5:26. https://doi.org/10.3389/fnins.2011.00026.

[74] Lammie C, Hamilton T, Azghadi MR. Unsupervised Character Recognition with a Simplified FPGA Neuromorphic System [Journal Article]. IEEE explore. 2018 May;p. 1–5. https://doi.org/10.1109/ISCAS.2018.8351532.

[75] Song S, Miller KD, Abbott LF. Competitive Hebbian learning through spike-timing-dependent synaptic plasticity [Journal Article]. Nature Neuroscience. 2000;3(9):919–926. https://doi.org/10.1038/78829.

[76] Shi X, Zeng Z, Yang L, Huang Y. Memristor-Based Circuit Design for Neuron With Homeostatic Plasticity. IEEE Transactions on Emerging Topics in Computational Intelligence. 2018 Oct;2(5):359–370. https://doi.org/10.1109/TETCI.2018.2829914.

[77] Lazar A, Pipa G, Triesch J. Fading memory and time series prediction in recurrent networks with different forms of plasticity. Neural Networks. 2007;20(3):312–322. Echo State Networks and Liquid State Machines. https://doi.org/https://doi.org/10.1016/j.neunet.2007.04.020.

[78] Mihalas S, Niebur E. A Generalized Linear Integrate-and-Fire Neural Model Produces Diverse Spiking Behaviors. Neural Computation. 2009;21(3):704–718. PMID: 18928368. https://doi.org/10.1162/neco.2008.12-07-680. https://arxiv.org/abs/https://doi.org/10.1162/neco.2008.12-07-680.

[79] Akiba T, Sano S, Yanase T, Ohta T, Koyama M. Optuna: A Next-Generation Hyperparameter Optimization Framework. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery; Data Mining. KDD '19. New York, NY, USA: Association for Computing Machinery; 2019. p. 2623–2631. Available from: https://doi.org/10.1145/3292500.3330701.

[80] Naous R, Al-Shedivat M, Salama KN. Stochasticity Modeling in Memristors. IEEE Transactions on Nanotechnology. 2016 Jan;15(1):15–28. https://doi.org/10.1109/TNANO.2015.2493960.

[81] Payvand M, Nair MV, Müller LK, Indiveri G. A neuromorphic systems approach to in-memory computing with non-ideal memristive devices: From mitigation to exploitation. Faraday Discussions. 2019;213:487–510.

[82] Rieck JL, Hensling FV, Dittmann R. Trade-off between variability and retention of memristive epitaxial SrTiO3 devices. APL materials. 2021;9(2):021110.

[83] Ford AJ, Jha R. Memristive device variability performance impact on neuromorphic machine learning hardware. In: 2020 11th International Green and Sustainable Computing Workshops (IGSC). IEEE; 2020. p. 1–7.