electronics

MDPI

Article

Exploring the Impact of Variability in Resistance Distributions of RRAM on the Prediction Accuracy of Deep Learning Neural Networks

Nagaraj Lakshmana Prabhu [1], Desmond Loy Jia Jun [2,3], Putu Andhita Dananjaya [2], Wen Siang Lew [2], Eng Huat Toh [3] and Nagarajan Raghavan [1,*]

[1] Engineering Product Development (EPD) Pillar, Singapore University of Technology and Design, 8 Somapah, Building 1, Level 3, Singapore 487372, Singapore; lakshmana_nagaraj@mymail.sutd.edu.sg

[2] Physics & Applied Physics Division, School of Physical and Mathematical Sciences, Nanyang Technological University, 21 Nanyang Link, Singapore 637371, Singapore; dloy002@e.ntu.edu.sg (D.L.J.J.); andhita1@e.ntu.edu.sg (P.A.D.); wensiang@ntu.edu.sg (W.S.L.)

[3] Globalfoundries, 60 Woodlands Industrial Park D Street 2, Singapore 738406, Singapore; enghuat.toh@globalfoundries.com

* Correspondence: nagarajan@sutd.edu.sg; Tel.: +65-6499-8756

Received: 19 October 2019; Accepted: 20 February 2020; Published: 29 February 2020

Abstract: In this work, we explore the use of the resistive random access memory (RRAM) device as a synapse for mimicking the trained weights linking neurons in a deep learning neural network (DNN) (AlexNet). The RRAM devices were fabricated in-house and subjected to 1000 bipolar read-write cycles to measure the resistances recorded for Logic-0 and Logic-1 (we demonstrate the feasibility of achieving eight discrete resistance states in the same device depending on the RESET stop voltage). DNN simulations have been performed to compare the relative error between the output of AlexNet Layer 1 (Convolution) implemented with the standard backpropagation (BP) algorithm trained weights versus the weights that are encoded using the measured resistance distributions from RRAM. The IMAGENET dataset is used for classification purpose here. We focus only on the Layer 1 weights in the AlexNet framework with 11 × 11 × 96 filters values coded into a binary floating point and substituted with the RRAM resistance values corresponding to Logic-0 and Logic-1. The impact of variability in the resistance states of RRAM for the low and high resistance states on the accuracy of image classification is studied by formulating a look-up table (LUT) for the RRAM (from measured *I-V* data) and comparing the convolution computation output of AlexNet Layer 1 with the standard outputs from the BP-based pre-trained weights. This is one of the first studies dedicated to exploring the impact of RRAM device resistance variability on the prediction accuracy of a convolutional neural network (CNN) on an AlexNet platform through a framework that requires limited actual device switching test data.

Keywords: convolutional neural network; look-up-table; resistive switching memory; synapse; variability

1. Introduction

The advent of convolutional neural networks (CNN) has brought about a paradigm shift in the adoption of hardware assisted deep learning for edge computing applications. Several non-volatile memory based device technologies have been explored to realize neuromorphic/deep learning based CNN systems including resistive switching memory (RRAM) [1,2], spin-transfer torque RAM (STT-RAM) [3,4] and phase-change memory (PCM) [5,6] for synaptic storage of the weights of the network,

which form the core of the computation. While many of these device technologies have been shown to be good candidates at an array level for hardware implementation, the issue of "*variability*" in the device performance and its impact on the reduction in prediction accuracy (say for image classification) of the neural network has not been specifically dealt with. In this study, we aim to propose a framework to assess and "quantify" the impact of RRAM switching variability on the prediction accuracy of CNN. The RRAM is chosen as the candidate device here as it has a simple structure [7], enables high integration density (small form factor) and can exhibit switching in the low-power regime, although the variability in the switching trends can be quite high [8–10].

2. Background and Literature Review

2.1. Overview of Convolutional Neural Networks

Convolutional neural networks (CNNs) are a subset of the machine learning family, which are well known for its grid-like topology, in which each grid is computed in parallel and hence best suited for a low footprint "digital signal processor" like implementation. The CNN computation can be explained with time-series data, by processing pixel data at regular time intervals. The CNN can be implemented effectively on a parallel processing system, in which the mathematical matrix operation of *convolution*, is followed by a fully connected neural network implementation [11]. The mathematical operation of "convolution" is illustrated in Figure 1. The kernel or trained weight is applied to the entire image, as shown in Figure 1. The boxes with arrows show how the input tensor element gets transformed to the output tensor by using the trained weight or kernel. Following the convolution, we have the fully connected layers that connect every neuron in one layer to every neuron in another layer, similar to the traditional multi-layer perceptron (MLP) neural network. Every neuron in the given network computes an output value by applying a function to the input values from the previous layer, as shown in Figure 2. The function applied to the input values is specified by trained weights. Learning in a neural network is a process of adjusting its weights (minimizing the cost function by iteration) [12].

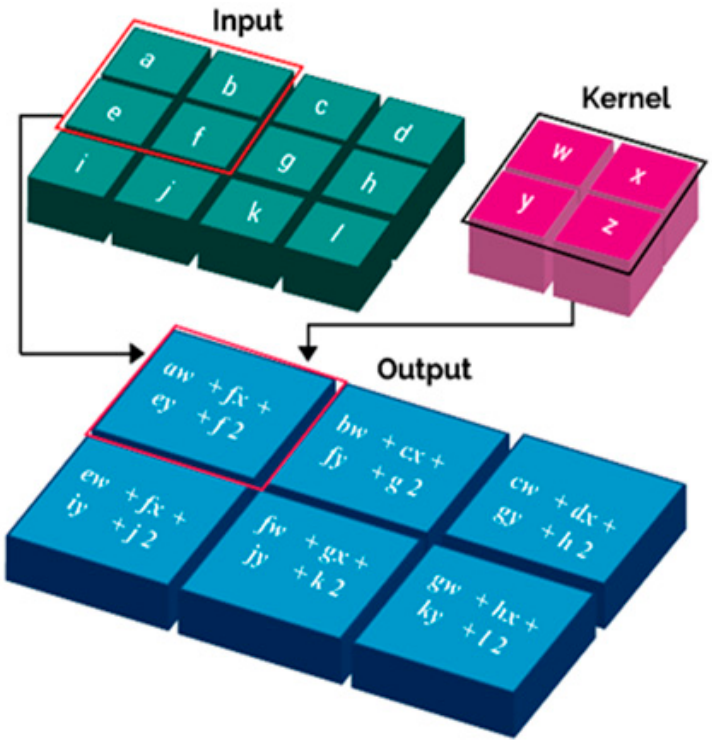

Figure 1. Illustrating the procedure of convolution in a convolutional neural network (CNN).

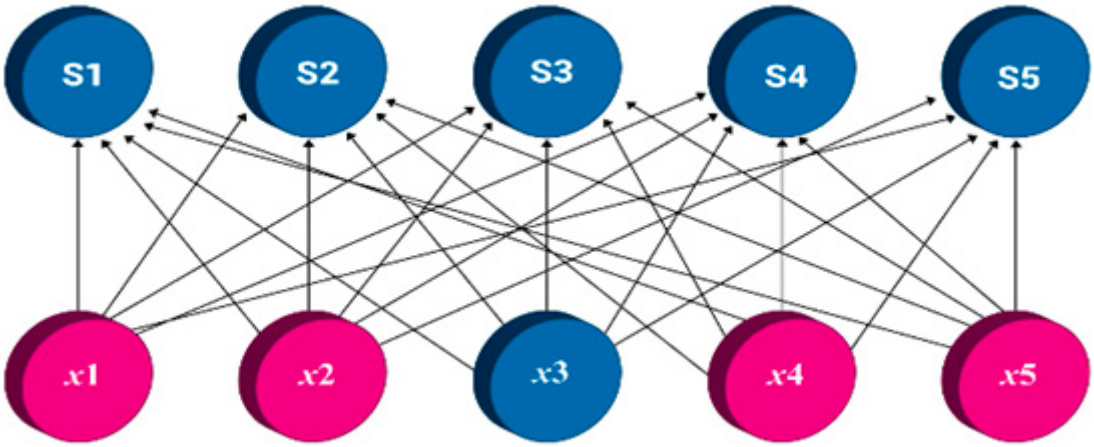

Figure 2. Part of the neural network architecture with a fully connected layer with input layer denoted by the nodes x1~x5 and the output layer denoted by S1~S5. Here, as an example, with matrix multiplication, all the outputs are affected by x3 (shown in blue).

2.2. Use of RRAM for Neuromorphics/Deep Learning

Resistive switching random access memory (RRAM) devices are suitable for neuromorphic applications as the change of the weights in the neural network can be mimicked by the analog conductance change in the dielectric of the RRAM, which changes state due to defect (vacancy) generation, transport or annihilation. The application of an electrical current or voltage as the stimuli creates an ionic motion or vacancy cluster formation in a nanoscale dimension within a two-terminal device leading to local redox phenomena and this, in turn, affects the device resistance, which stays non-volatile, unless perturbed externally again by applying another voltage or current pulse. Devices exhibiting such properties are, in general, termed as memristors. Figure 3 depicts the typical current-voltage (*I-V*) characteristics of an RRAM, which could be switched either in the bipolar mode or unipolar mode (only bipolar schema shown here for illustration). The evolution from high to low resistance state is called SET, while the transition from the low to high resistance state is referred to as RESET.

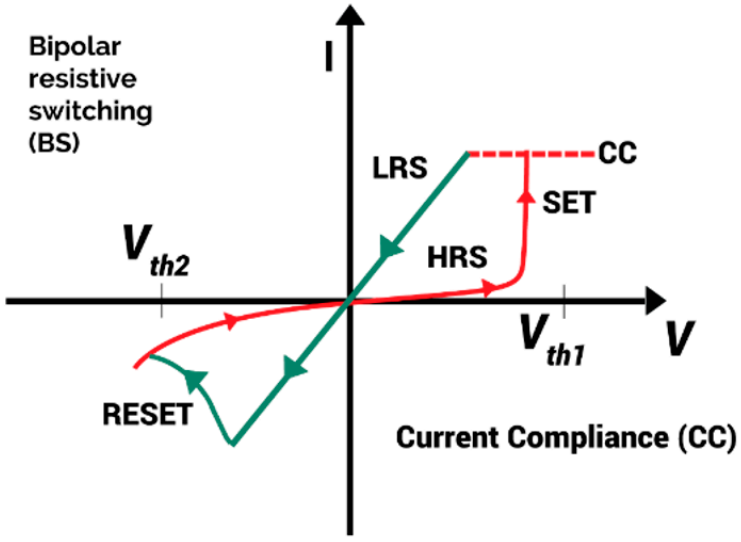

Figure 3. Typical *I-V* plot for the bipolar switching trends in a resistive switching memory device. The image here is adopted and modified from Reference [13], Copyright 2015, John Wiley & Sons.

The reason for RRAM devices to be used in neuromorphic application are its analog conductance changes (depending on the material stack and operating conditions), high scalability, ultrafast

switching speed, non-volatility, large HRS (High Resistance State) /LRS (Low Resistance State) window, non-destructive read, simple structures with standard materials, 3D stackability and multi-level storage (the option to store more than one bit of information in one cell). Low write current (short electrical pulses able to change the resistive state), longer retention time and endurance are other factors that favor the adoption of RRAM. Lower operating energy per bit and excellent CMOS (Complementary Metal Oxide Semiconductor) process compatibility and manufacturability are also advantages of using the RRAM as the primary element for neuromorphic computing [1].

2.3. Use of the AlexNet Platform for Implementation

AlexNet, which is one of the popular CNN, competed in the ImageNet Large Scale Visual Recognition Challenge in 2012, with an error percentage of 15.3%. The entire model was computationally expensive, as it was implemented with a graphics processing units (GPUs), making the network perform well in real time. Figure 4 shows the schematic of the AlexNet containing eight layers out of which five are convolutional and three are fully-connected layers. The first convolutional layer filters the 224 × 224 × 3 input image with 96 kernels of size 11 × 11 × 3 with a stride of 4 pixels (this is the distance between consecutive computations). The second convolutional layer takes as input, the output of the first convolutional layer and filters it with 256 kernels of size 5 × 5 × 48. The third, fourth, and fifth convolutional layers are connected to one another without any intervening pooling or normalization layers. The third convolutional layer has 384 kernels of size 3 × 3 × 256 connected to the (normalized, pooled) outputs of the second convolutional layer. The fourth convolutional layer has 384 kernels of size 3 × 3 × 192 and fifth convolutional layer has 256 kernels of size 3 × 3 × 192. The fully-connected layers have 4096 neurons each [11]. While the combination of AlexNet and RRAM for deep learning implementation is not new and have been proposed in References [14–18] and these works did not give any details on the simulation methodology and procedure. Our work here provides a full system flow chart that explains how the variability in RRAM was explored and exploited to represent the "binary" representation of the weights in the CNN.

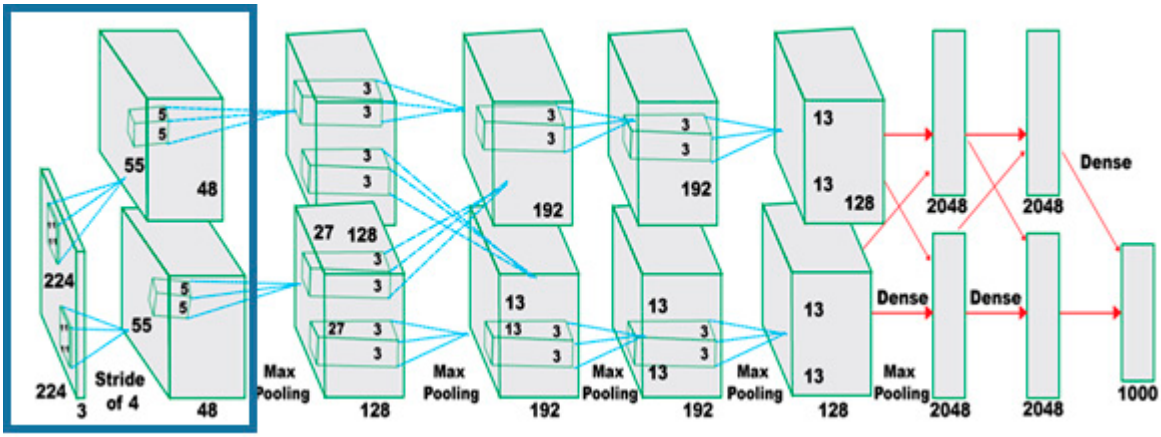

Figure 4. Layer 1 is a convolution layer, the input image size is—224 × 224 × 3 (3 = RGB (8 bits each)), number of filters → 96 (Filter size—11 × 11 × 3) and the layer 1 output comprises of (224/4 × 224/4 × 96 = 55 × 55 × 96) (with a stride of 4 pixels).

2.4. Scope and Outline of Study

This paper is organized as follows. Section 3 explains the device fabrication of RRAM and the simulation framework setup for the performance analysis of the CNN where the weights in the AlexNet Layer 1 are represented with the RRAM LUT (look-up table). Section 4 presents the electrical characterization results of the fabricated RRAM device, resulting in seven different HRS states depending on the magnitude of the RESET stop voltage pulse. In Section 5, we discuss the simulation results obtained from embedding RRAM conductance variations into the AlexNet Layer 1 weights.

Finally, we conclude our study in Section 6 revisiting the novelty of our study and proposing possible recommendations for further research along this line.

3. Experimental and Simulation Test Setup

3.1. HfO_x RRAM Device Fabrication

The HfO_x-based RRAM (1R) devices were fabricated on an SiO_2 substrate, as shown in Figure 5. The bottom electrodes were patterned by UV lithography with positive tone resist followed by sputtering of the inert 10 nm Pt metal electrode under Ar ambient at room temperature and with a lift-off process. The process was repeated to sputter 15 nm HfO_x layer and then the active top metal electrode of Ti (10 nm) to form 10 × 10 µm crosspoint devices. Finally, another 10 nm of inert Pt layer was deposited on top of the Ti layer for passivation. The *I-V* curves and endurance cycles were characterized using the Keithley SCS-4200 A system. The SET operation was initiated using 3 V, 5 ms pulses with an external current compliance setting of 100µA, while the RESET events were triggered by stepwise increase in pulse amplitudes with an opposite voltage polarity (from −1.2 V to −2.2 V with 0.2 V intervals) with a pulse period of 200 ns to achieve multilevel resistance states.

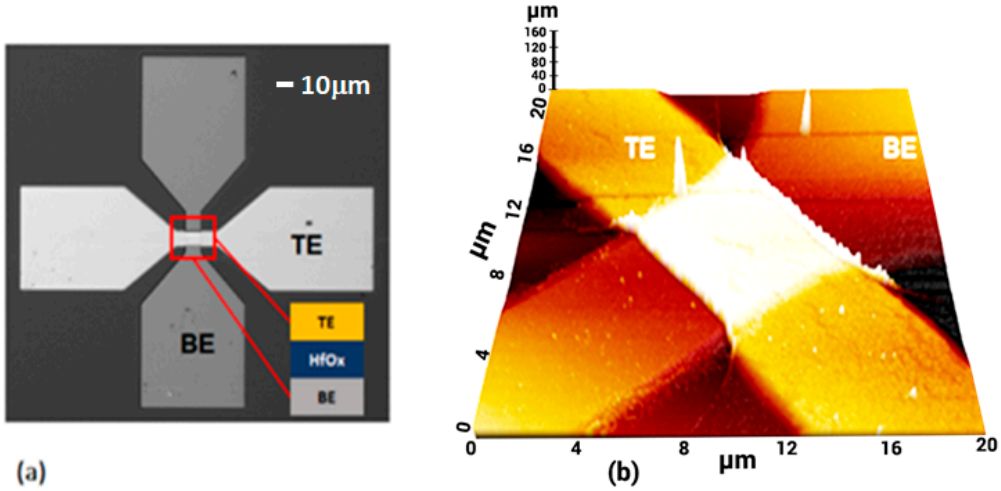

Figure 5. (**a**) Top view of the cross-point device under an optical microscope and a cross-sectional view of the schematic of the structure in the inset figure. The bottom electrode (BE) consists of an inert electrode, and top electrode (TE) serves as the active electrode, which essentially functions as an oxygen reservoir to facilitate oxygen migration in and out of the oxide layer. This allows the switching behavior to be predominantly governed by oxygen ion movement instead of Joule heating. (**b**) Three-dimensional (3D) view of the cross-point region (see red rectangle) of the structure, taken using an atomic force microscope.

3.2. RRAM Based Synaptic Simulation Setup

The framework for RRAM resistance distribution incorporation into the AlexNet Layer 1 using a look-up table (LUT) approach is illustrated in Figure 6. The AlexNet Layer 1 convolution operation is implemented in MATLAB®, with two pipelines, one which uses AlexNet's original extracted trained weights (96 filters each of 11 × 11 for red-green-blue (RGB)), implemented with MATLAB® general matrix and the second pipeline, which constructs the AlexNet weights from the LUT. The LUT contains the algorithm trained synaptic weight in a binary format, 32-bit (24-bits for the mantissa, 7-bits for exponent and 1-bit for sign) for every single weight value, as illustrated in Figure 7. Each algorithm trained bit, Logic-0 or Logic-1, in the trained weight, is substituted by resistance values from the low and high resistance states of the switching RRAM device, encoded into a Logic-0 or Logic-1, based on a set resistance threshold level (R_{TH}). The difference in the accuracy of prediction of the Layer-1 computed output between the two pipelines provides a quantitative

estimate of the variability induced error in the hardware implementation of the CNN, in comparison to its algorithmic counterpart.

The detailed step-by-step implementation of the RRAM based CNN setup is described below, along with the approach to quantify and compare the prediction errors.

Step 1 A 224 × 224 pixel sized "cat" image is taken as the input. The program reads the image in .jpg format, converts it to uncompressed RGB, and this computed data is then passed to the next step for further processing.

Step 2 The RGB image (224 × 224 × 3) is copied multiple times with a window size of 11 × 11 each and with a 4-pixel difference between the adjacent windows. The output from this stage is a matrix of size 11 × 11 × 3025 for each color (RGB).

Step 3 There are two parallel pipelines here—the first (MATLAB® array) reads the weights (which are software trained using backpropagation) from a local drive to the program cache for further processing. The second pipeline (RRAM LUT) represents these software trained weights in a floating point format and encodes these weights with a Logic-0 or Logic-1 using the measured resistance distribution of RRAM in the low and high resistance states based on comparison of the randomly sampled resistance value from the LRS/HRS best fit resistance distributions to a set threshold resistance value (R_{TH}). The resulting floating point representation of the RRAM encoded weights is then passed onto the next stage. The output size of both pipelines is 11 × 11 × 96 (filters) × 3(RGB). It is the second pipeline where the physical variability (measured by resistance of low (LRS) and high resistance states (HRS)) inherent in the fabricated RRAM device is incorporated into the CNN.

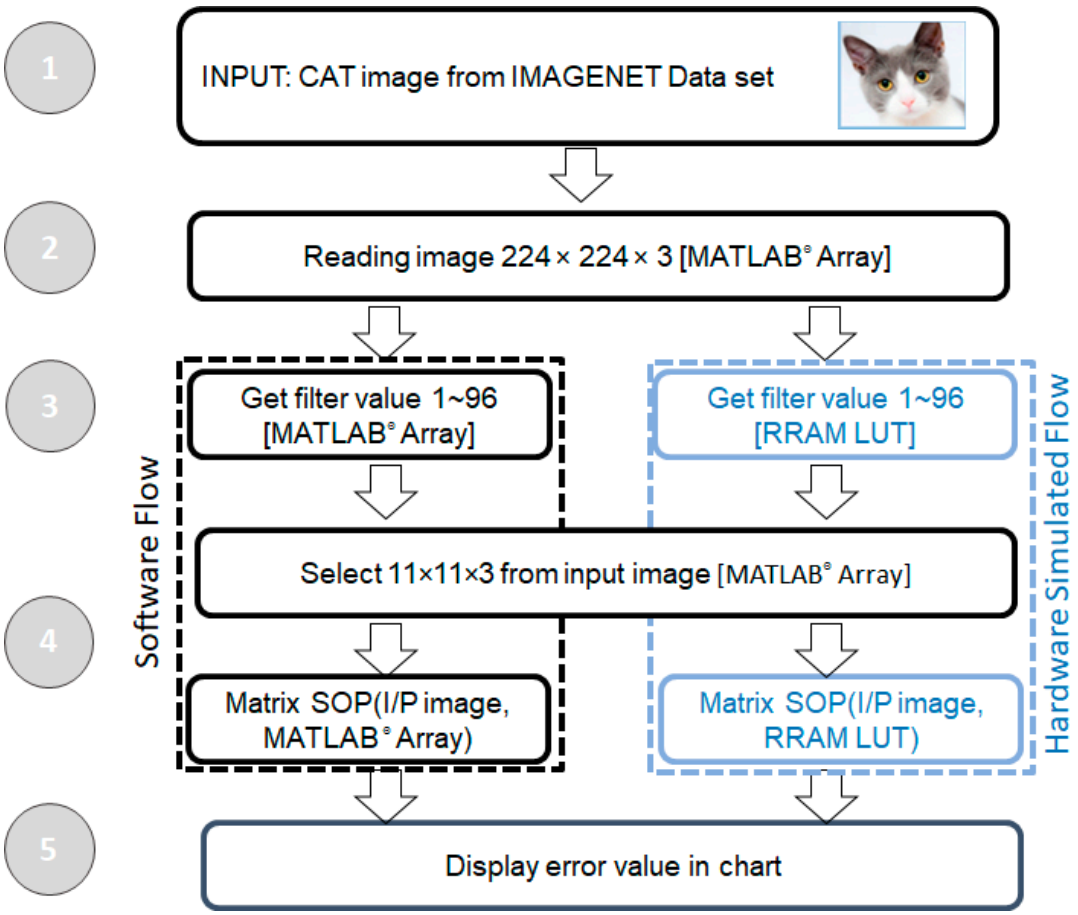

Figure 6. Flowchart showing the implementation of resistive random access memory (RRAM) based look-up table (LUT) for error quantification of a hardware implementation of the CNN using the AlexNet schema with Layer 1 being the convolution layer where the LUT is incorporated.

Step 4 For each 11 × 11 window of the input image (total 3025), the matrix SOP (sum of the product) is separately computed using the weights from the two pipelines. The difference in the computed output for the two pipelines is then recorded and defined as the "prediction error".

Step 5 To capture the stochastics of the error in computation, Steps 3 and 4 are iterated 1000 times using different randomly sampled resistance values for the HRS and LRS states of the RRAM device.

3.3. Generation of RRAM based LUT for AlexNet Weights

As illustrated in Figure 7, the software trained weights are obtained based on the stochastic gradient descent (SGD) optimization algorithm, optimized for one million images over 1000 categories. Each of these trained weights are represented in a 32-bit floating point format comprising of 24 bits of mantissa, 7 bits of exponent, and one sign bit. Every single matrix element within a filter within a certain layer of the AlexNet framework would need to represented using this 32-bit format.

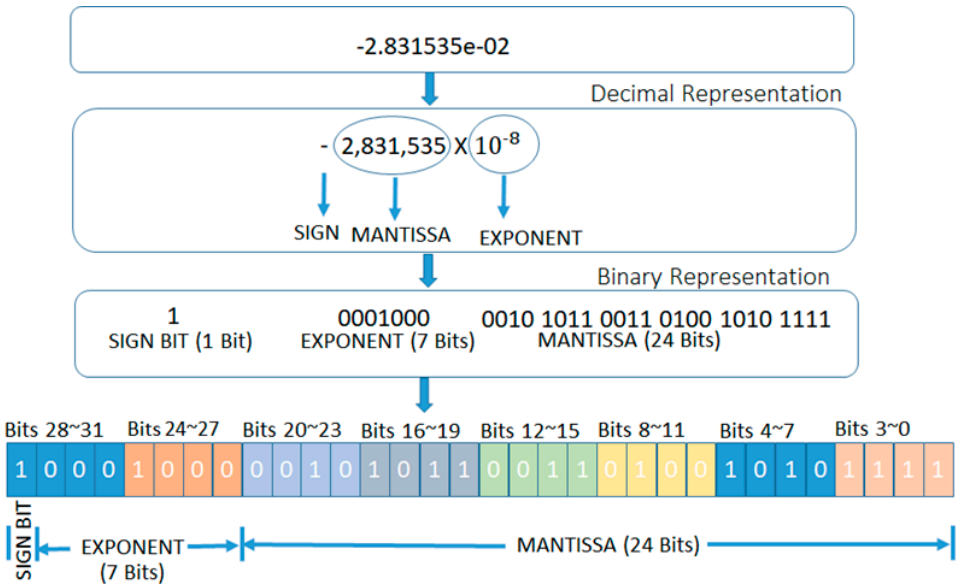

Figure 7. RRAM weights representation in binary (Floating points values).

Figure 8 illustrates how the measured electrical resistance of the RRAM is incorporated into the logic-0/1 for every single bit of the 32-bit representation of the algorithm trained weight in Figure 7. The DC *I-V* characteristics of the HfO_x-based RRAM device with abrupt SET and gradual RESET data are measured and the corresponding resistances in the LRS and HRS are extracted at a specific read voltage (V_{READ} ~ 150 mV in our case). The extracted resistances in the HRS and LRS states (R_{HRS} and R_{LRS}) are then fitted with a Lognormal distribution. As seen in Figure 8, the extracted resistance distributions are expected to overlap at the tail ends which is where the bits in the floating point could be erroneously represented in the CNN hardware implementation.

To demarcate the "0" and "1" from the resistance distributions, a threshold resistance (R_{TH}) is arbitrarily defined by the average of the log scale values of the mean resistances in the HRS and LRS states (as shown in the equation below). For any randomly sampled value (R) from the fitted HRS or LRS distribution, if $R < R_{TH}$, the value is encoded as a "0"; else, it is taken as "1". For every bit in the floating point representation for error computation in Step 5 of Figure 6, 1000 values of R_{HRS} and R_{LRS} are sampled from their distributions and encoded to Logic-0/Logic-1 using the rule below.

$$log\ R_{TH} = \frac{log\ R_{LRS} + log\ R_{HRS}}{2}$$

If $R < R_{TH}$ = *logic 0*

Else If R > R_{TH} = logic 1

In this study, the LRS and HRS states are denoted as Logic-0 and Logic-1 respectively. Sampled values of R_{LRS} from the LRS distribution that have a value larger than R_{TH} will be wrongly classified as a "1". Similarly, samples values of R_{HRS} from the HRS distribution will get classified as a "0" if the value is lower than R_{TH}. These incorrectly coded bits are referred to here as "False-0" and "False-1", as shown in Figure 8.

The above LUT generation procedure using MATLAB® subroutine is applied for four inner loops as discussed in the following subsection.

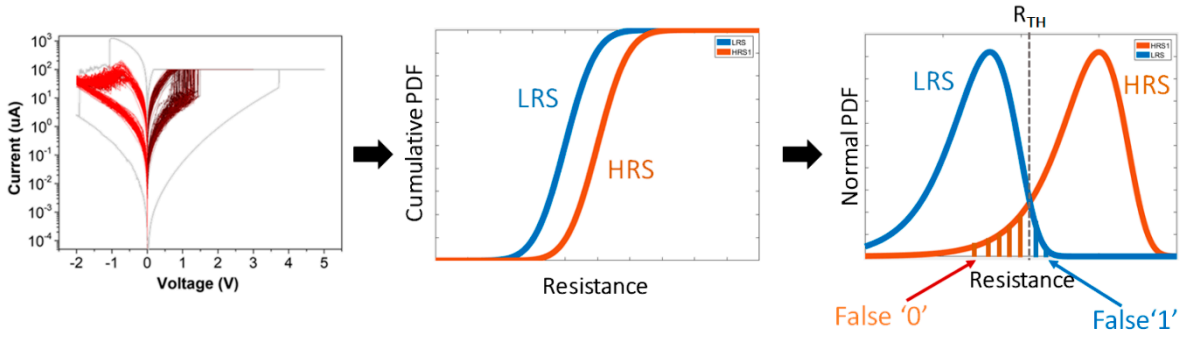

Figure 8. RRAM resistance coded to digital logic-0/1 with false logic-0/1 condition explained on a normal pdf plot of low resistant state (LRS) and high resistant state (HRS) resistance data.

3.4. Multiple Embedded Loops of CNN Simulation

To fully examine the impact of hardware device variability on the accuracy drop in the CNN implementation for a practical edge computing application, the following four loops of simulation are deemed necessary:

Loop 1 As illustrated in Figure 9, Loop-1 involves replacement of a group of just 4 bits every time from the 32-bit string representation of the algorithm trained weight with the RRAM encoded binary values. The binary values of the remaining 28 bits are intentionally undisturbed. With this 4-bit perturbation from the least significant bit (LSB) to the most significant bits (MSB) (Bits3~0, Bits4~7, Bits8~11, Bits12~15, Bits16~19, Bits20~23, Bits 28~31), the impact of the bit position on the prediction error in a hardware CNN can be explicitly quantified.

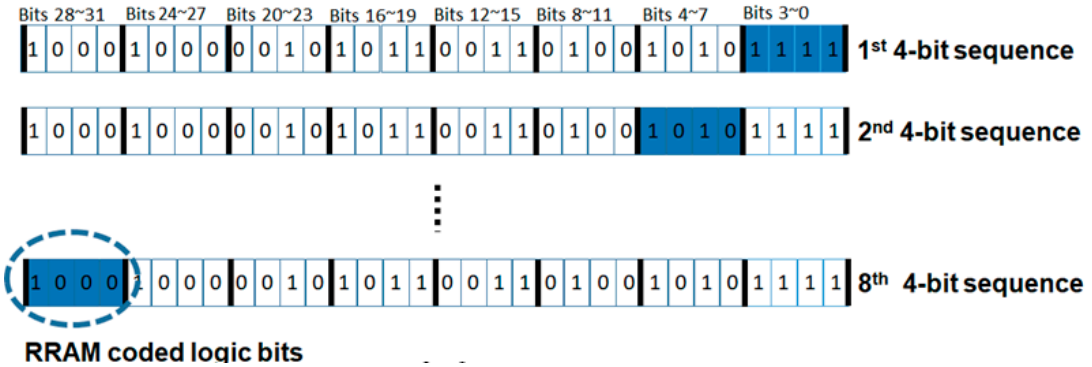

Figure 9. The blue-shaded 4-bit string represents the bits where RRAM resistance values are encoded and the remaining 28 bits represent the unchanged binary version of the algorithm trained weight.

Loop 2 Depending on the RESET stop voltage (VRESET-STOP), the resistance of the HRS state exhibits a wide range of values. For the same LRS state (governed by the current compliance), the RRAM device is reset to different HRS states by tuning VRESET-STOP. Seven different HRS states were attained by a selective tuning of VRESET-STOP from −1.2 V to −2.2 V, keeping the pulse duration fixed at 200 ns. For each of the resulting HRS distributions (shown in Figure 10),

represented by HRS-1 (shallow reset) to HRS-7 (deep reset), a different LUT was generated. This loop allows the quantification of the memory window on the CNN prediction error given the increasing resistance variability at deeper reset conditions.

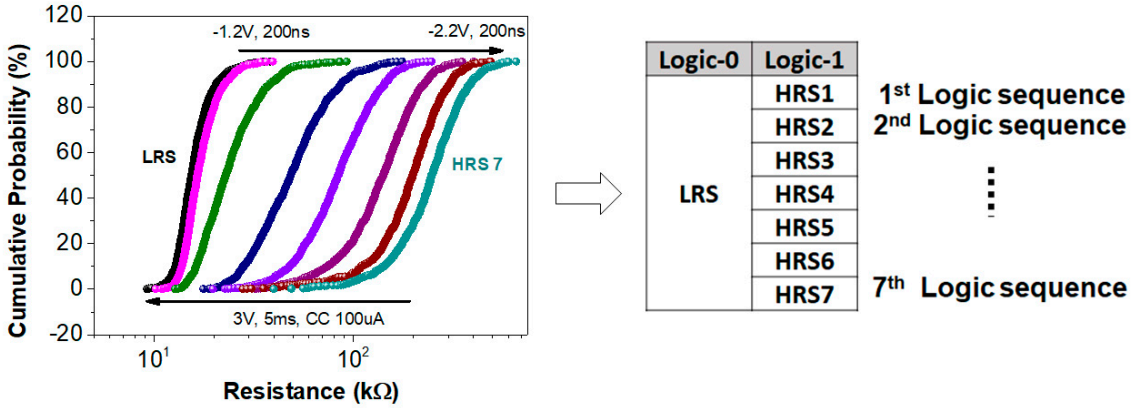

Figure 10. The coding scheme consisting of 7 different HRS states (one at a time) and 1 LRS state for the LUT generation. Attaining different reset conditions allows for the examination of memory window impact on the prediction error, also accounting for the higher resistance variability at deeper reset conditions.

Loop 3 The upper and lower pipeline of the AlexNet Layer-1 contains 96-trained filters in total, and each of them are sized with a 11 × 11 matrix (refer to Figure 11). Loop operations 1 and 2 above are repeated for all the 96-filter values to generate RRAM resistance encoded weights the entire Layer-1 for a full convolution operation.

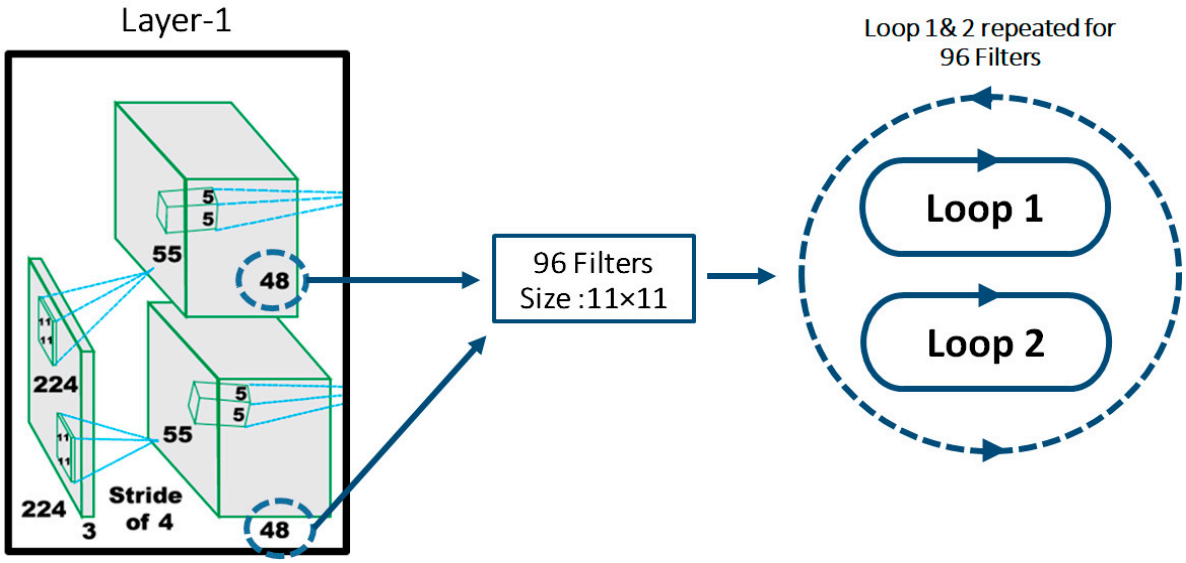

Figure 11. Total of 96 AlexNet Layer-1 filters with 48 filters each from the upper and lower pipeline.

Loop 4 The above loops 1, 2 and 3 are then repeated for red, blue, and green pixel image data, respectively, as shown in Figure 12.

In all, simulation of all the scenarios encompassing the above four inner loops results in a total of 1,951,488 iterations from the 1 K bits of RRAM cycle-to-cycle resistance switching data measured. Every single algorithm trained weight value is converted to 56 different binary combination sequences (containing 8 4-bit string patterns × 7 HRS state values = 56 different LUT data sets).

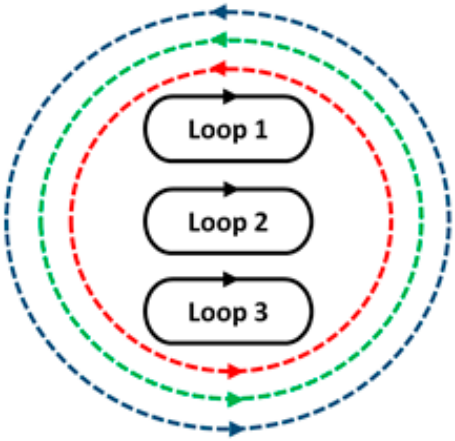

Figure 12. Iteration repeated for red, green and blue data of the pixels in the classified image pattern.

4. Electrical Characterization Results and Discussion

The *I-V* characteristic of the fabricated RRAM device is shown in Figure 13a for 100 switching cycles. It is apparent that while the bipolar switching is gradual and consistently observed, the *I-V* curves for the different HRS states show a significant overlap. While such substantial overlap in the resistance values is undesirable from a CNN hardware implementation point of view, the experimental data sets are being used here only to illustrate the methodology that is proposed in Section 3. With further optimization of the device stack, fabrication process, and operating conditions, it should be possible to realize more confined and distinguishable resistance patterns in the future. The current measured at a read voltage of V_{READ} ~ 150mV for a single tested device is plotted in Figures 13b and 14a after every SET and RESET sweep, keeping the pulse period fixed at 200 ns. The corresponding cumulative density function plots of the 8 resistance states (in terms of conductance) are plotted in Figure 14b. Multiple resistance states are attained here by gradually increasing the magnitude of the RESET pulses from −1.2 V to −2.2 V at steps of −0.1 V. The gradual process of RESET in this device stack may enable the realization of a multi-bit cell in the future if the stack is further optimized to ensure more spread out and discrete patterns of resistance variation with minimal overlap.

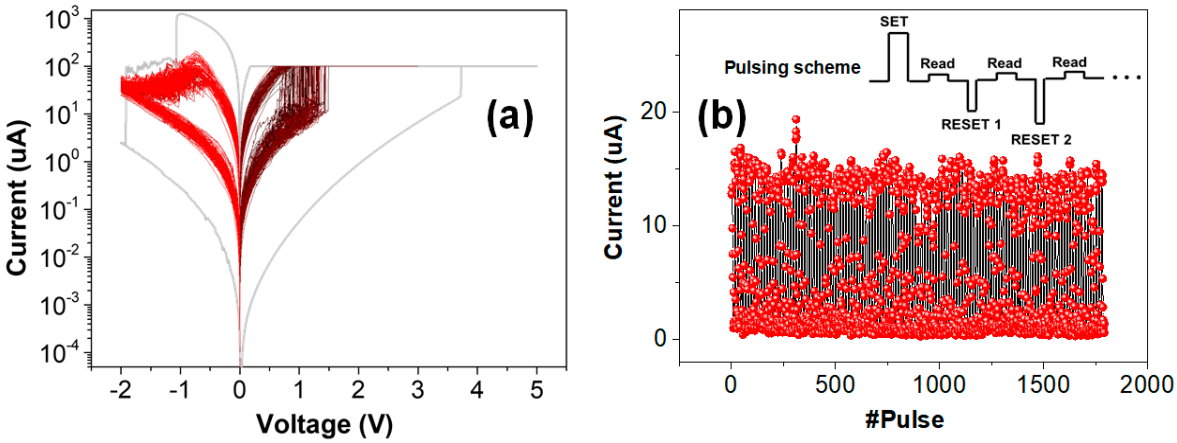

Figure 13. (**a**) DC *I-V* characteristics of HfO_x-based RRAM device with abrupt SET and gradual RESET behavior. The grey curve indicates the forming process on the pristine device. (**b**) Endurance cycling performed with a single SET pulse and multiple RESET pulses (−1.2 V to −2.2 V with 0.2 V interval) for each cycle.

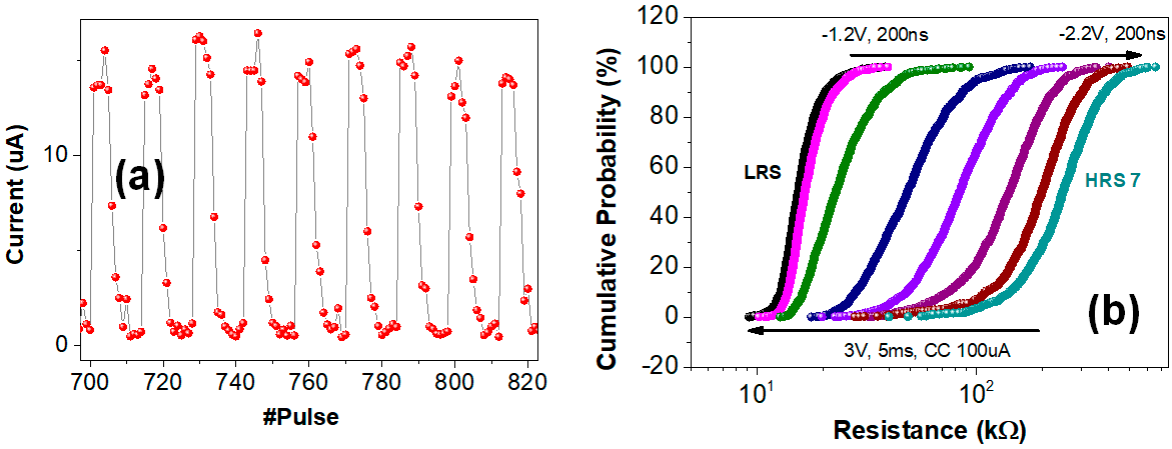

Figure 14. (**a**) Closer look at the current jumps during the endurance test from the 700th to 820th pulse cycle. (**b**) Cumulative probability of the resistance states (1 LRS and 7 HRS states) achieved by successively increasing RESET voltage pulse amplitudes (−1.2 V to −2.2 V) during the multi-step RESET process, based on the pulsing scheme shown in the inset of Figure 13b

5. Simulation Results and Discussion

This section examines the results of the simulation of the AlexNet CNN layer-1 involving computation of the output matrix SOP for the two different pipelines shown in Figure 15 (reproduced from Figure 6). The value of matrix SOP from the purely algorithm trained weights is denoted as S_{CONV}, while the corresponding value from RRAM encoded LUT is denoted as H_{CONV}. We introduce a term called the *relative error (RE)* defined as the modulus of the difference between H_{CONV} and S_{CONV}, normalized by the value of S_{CONV}.

Figure 16 lists the values of *RE* for the R-G-B data considering different 4-bit positions along the 32-bit string and also the different HRS levels (HRS-1 corresponding to shallow reset all the way to HRS-7 for deep reset) attained by fine tuning $V_{RESET\text{-}STOP}$. When examining the impact of the bit positions where RRAM variability is incorporated, we see an increase of more than two orders of magnitude in the *RE* going from B3–B0 to B31–B28. This steep increase in error is very much expected given the increasing significance of the numerical values of the bits as we move from the right to the left in the 32-bit string. Note that the value of *RE* is substantially high for B31–B28 as the sign and exponent bits plays a critical role there.

When comparing the impact of the different HRS levels, a notable decrease in *RE* when we transition from HRS-1 to HRS-7 is only observed for B31–B28, but not for the other bit positions. By right, similar reduction in *RE* should be observed irrespective of the bit positions analyzed. These expected trends are obscured by the large overlaps in the resistance distributions of the different HRS states in our current fabricated device. When a more robust device with distinctive HRS states is demonstrated in the future, the reduction in *RE* will be apparent for all bit positions though the magnitude of reduction will vary.

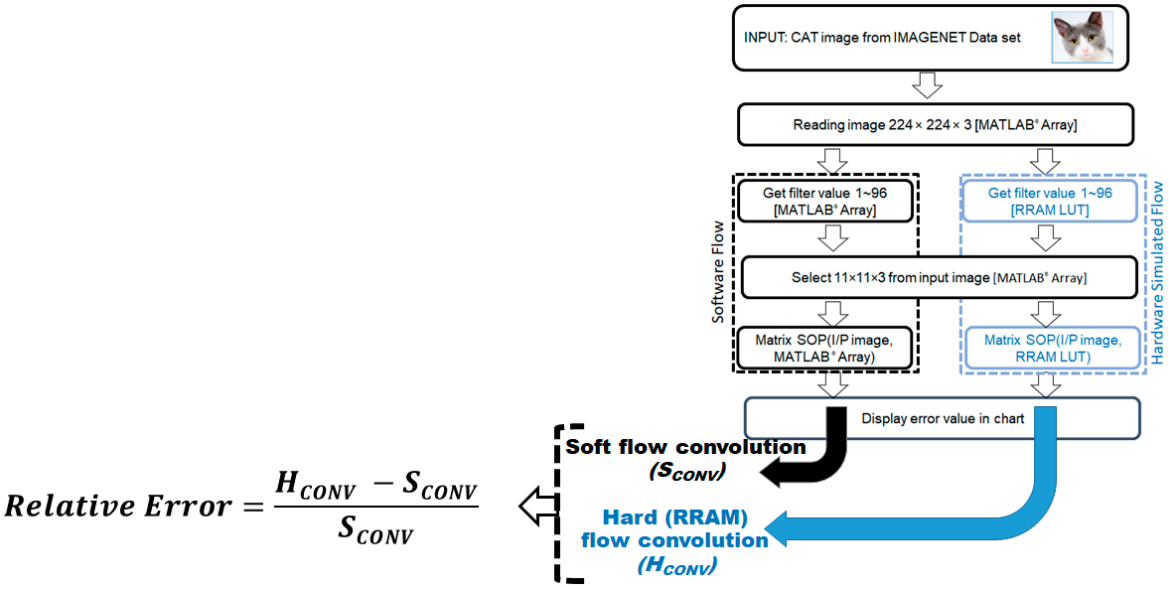

Figure 15. Definition of the relative error (RE) in computed matrix sum of products for the software and hardware flow convolution pipeline. The software pipeline refers to the algorithm trained weights while the hardware pipeline refers to the binary data encoded from the measured RRAM resistance distributions.

		AlexNET Layer 1 output: Relative Error						
		HRS1	***HRS2***	***HRS3***	***HRS4***	***HRS5***	***HRS6***	***HRS7***
		RED DATA						
BIT POSITION	***B3-0***	0.037	0.037	0.036	0.037	0.037	0.036	0.036
	B7-4	0.8	0.75	0.74	0.76	0.76	0.69	0.58
	B11-8	1.2	1.23	1.14	1.01	0.98	1.13	0.9
	B15-12	2.4	2.4	2.41	2.38	2.4	2.39	2.38
	B19-16	2.6	2.61	2.62	2.59	2.58	2.63	2.61
	B23-20	2.9	2.94	2.95	2.89	2.95	2.95	2.91
	B27-24	3.3	3.2	3.3	3.4	3.21	3.4	3.32
	B31-28	12	12	12.34	11.8	12	11.19	9.8
		GREEN DATA						
BIT POSITION	***B3-0***	0.035	0.037	0.04	0.038	0.036	0.039	0.041
	B7-4	0.84	0.79	0.74	0.77	0.78	0.71	0.62
	B11-8	1.23	1.25	1.17	0.96	1.03	1.17	0.89
	B15-12	2.35	2.42	2.39	2.4	2.43	2.41	2.4
	B19-16	2.55	2.61	2.63	2.55	2.63	2.62	2.65
	B23-20	2.91	2.93	2.94	2.9	2.93	3	2.92
	B27-24	3.31	3.25	3.27	3.41	3.28	3.42	3.37
	B31-28	11.17	12.02	12.32	11.83	12.05	12.03	9.85
		BLUE DATA						
BIT POSITION	***B3-0***	0.04	0.038	0.039	0.042	0.04	0.038	0.041
	B7-4	0.85	0.8	0.78	0.8	0.76	0.82	0.6
	B11-8	1.23	1.21	1.19	1.06	0.96	1.15	0.95
	B15-12	2.33	2.4	2.43	2.38	2.38	2.44	2.39
	B19-16	2.65	2.6	2.63	2.62	2.6	2.65	2.63
	B23-20	2.92	2.92	2.95	2.87	2.9	3.01	2.97
	B27-24	3.34	3.25	3.29	3.4	3.25	3.35	3.35
	B31-28	12	12	12.29	11.89	11.95	11.24	9.85

Figure 16. Bitwise relative error for all the 7-HRS values listed for different 4-bit positions along the 32-bit binary representation of the floating-point value for the CNN weights. The error values are listed separately for the red, green and blue colors accordingly.

6. Conclusions

This study presented a comprehensive methodology of assessing the impact of variability in the RRAM resistance distributions for the low and high resistance states on the image classification error

based on the synaptic weight representation using the 32-bit format for the floating point, with 4 bits at any time taking the values from the hardware implementation, while the remaining 28 bits having values from the pre-trained AlexNet CNN framework. A significant increase in the relative error (of the computed matrix SOP) from the least significant bits (LSB) to the most significant bits (MSB) is observed. The error value is particularly high for the MSB as it carries the exponent and the sign bit of the weight. It is also evident from the CNN simulations that the ability to reach deeper reset states more consistently also enables a significant reduction in *RE*.

The proposed LUT-based analysis proves to be a useful technique when there is a need for quick validation of the RRAM device performance and its impact on a large scale CNN network through simulation, rather than having to fabricate a large array of devices to quantify the actual loss in image classification accuracy. In general, device engineers fabricate only a handful of devices for a specific suite and combination of process parameters. They use these small arrays of RRAM devices to measure its electrical characteristics and switching performance for a few 100–1000 cycles. These small array of RRAM samples are certainly not sufficient to construct a fully functioning deep learning (DL) hardware platform. Considering the expenses and time taken for a full-fledged array level fabrication and the needed characterization setup (more so at academic institutions with limited facilities), it is vital for device engineers to be able to adopt a "short cut" approach to assess and validate their device performance in a practical CNN scale setting with minimal effort from a simulation framework point of view. This is where the proposed LUT-based framework here comes in handy.

One can use a small array of RRAM device switching data, extract the switching resistance state distributions, fit them with a Lognormal model and then generate a large LUT based data set and test the performance of their device for a specific deep learning application quickly by plugging the LUT into the MATLAB® simulation framework available for Deep Learning CNN. These quick simulations allow one to effectively quantify the expected prediction accuracy that can be attained for the fabricated devices. Our framework serves as a good intermediary step to assess whether to proceed with the existing process flow for a full array fabrication or to go back and optimize the RRAM process to attain better switching distributions. Our approach clearly enables one to quantify the impact of different memory windows and variations (and also different number of states (4-bit/8-bit etc.) on the image classification error using a simple set of devices (it could just be isolated set of MIM capacitors that were measured).

There are several possible improvements to be considered in subsequent studies. The current test setup arbitrarily defines a fixed threshold to classify any samples resistance value as Logic-0 or Logic-1. It may be worth exploring different definitions of the threshold levels to examine the changes in the classification errors. As discussed earlier, the fabricated stack in this study shows large variations in the resistance that result in significant overlap of the HRS and LRS distributions. The stack will be further optimized to improve the memory window, either by fabrication, choice of different switching voltages/pulse waveforms and/or through material engineering. Additionally, the impact of read disturb, random telegraph noise as well as endurance/retention degradation on the HRS/LRS distributions and their indirect influence on classification accuracy of a real scale CNN will also be a subject of study in the near future.

Author Contributions: N.L.P developed the concept and implemented the entire simulation framework. D.L.J.J fabricated the devices. P.A.D characterized the devices. W.S.L and N.R administered and supervised the project. E.H.T extended high-end fabrication support for device fabrication from the industry side. N.L.P and N.R. wrote the manuscript and N.R supervised the overall project and assisted in reviewing and editing the article. All figures and tables and illustrations were prepared by N.L.P along with support from D.L.J.J and P.A.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by A*STAR BRENAIC Research Project No. A18A5b0056 and the APC associated with the publication as well. Funding support for fabrication and characterization of devices were provided by the Economic Development Board EDB-IPP (RCA – 16/216) program and the Industry-IHL Partnership Program (NRF2015-IIP001-001).

Acknowledgments: The first author would like to thank the Ministry of Education (MOE), Singapore for providing the research student scholarship (RSS) at SUTD for 2018-2021. The authors from NTU would like to acknowledge the support provided by the Economic Development Board EDB-IPP (RCA – 16/216) program and the Industry-IHL Partnership Program (NRF2015-IIP001-001). Prof. Wen Siang Lew (NTU) is a member of the Singapore Spintronics Consortium (SG-SPIN) and acknowledges their support as well. Prof. Nagarajan Raghavan (SUTD) would like to acknowledge the financial and logistical support from the A*STAR BRENAIC Research Project No. A18A5b0056, which enabled the work to be accomplished.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Park, S.; Kim, H.; Choo, M.; Noh, J.; Sheri, A.; Jung, S.; Seo, K.; Park, J.; Kim, S.; Lee, W.; et al. RRAM-based synapse for neuromorphic system with pattern recognition function. In Proceedings of the 2012 International Electron Devices Meeting, San Francisco, CA, USA, 10–13 December 2012.
2. Yu, S.; Gao, B.; Fang, Z.; Yu, H.; Kang, J.; Wong, H.S.P. A neuromorphic visual system using RRAM synaptic devices with sub-pJ energy and tolerance to variability: Experimental characterization and large-scale modeling. In Proceedings of the 2012 International Electron Devices Meeting, San Francisco, CA, USA, 10–13 December 2012.
3. Vincent, A.F.; Larroque, J.; Locatelli, N.; Romdhane, N.B.; Bichler, O.; Gamrat, C.; Zhao, W.S.; Klein, J.O.; Galdin-Retailleau, S.; Querlioz, D. Spin-transfer torque magnetic memory as a stochastic memristive synapse for neuromorphic systems. *IEEE Trans. Biomed. Circuits Syst.* **2015**, *9*, 166–174.
4. Zhang, D.; Zeng, L.; Qu, Y.; Wang, Z.M.; Zhao, W.; Tang, T.; Wang, Y. Energy-efficient neuromorphic computation based on compound spin synapse with stochastic learning. In Proceedings of the 2015 IEEE International Symposium on Circuits and Systems (ISCAS), Lisbon, Portugal, 24–27 May 2015.
5. Suri, M.; Bichler, O.; Querlioz, D.; Cueto, O.; Perniola, L.; Sousa, V.; Vuillaume, D.; Gamrat, C.; DeSalvo, B. Phase change memory as synapse for ultra-dense neuromorphic systems: Application to complex visual pattern extraction. In Proceedings of the 2011 International Electron Devices Meeting, Washington, DC, USA, 5–7 December 2011.
6. Garbin, D.; Suri, M.; Bichler, O.; Querlioz, D.; Gamrat, C.; DeSalvo, B. August. Probabilistic neuromorphic system using binary phase-change memory (PCM) synapses: Detailed power consumption analysis. In Proceedings of the 2013 13th IEEE International Conference on Nanotechnology (IEEE-NANO 2013), Beijing, China, 5–8 August 2013.
7. Dutta, M.; Maikap, S.; Qiu, J.T. Controlled Conductive filament and Tributyrin Sensing Using an Optimized Porous Iridium Interfacial Layer in Cu/Ir/TiN_xO_y/TiN *Adv. Electron. Mater.* **2019**, *5*, 1800288.
8. Fantini, A.; Goux, L.; Degraeve, R.; Wouters, D.J.; Raghavan, N.; Kar, G.; Belmonte, A.; Chen, Y.Y.; Govoreanu, B.; Jurczak, M. Intrinsic switching variability in HfO_2 RRAM. In Proceedings of the 2013 5th IEEE International Memory Workshop, Monterey, CA, USA, 26–29 May 2013.
9. Goux, L.; Fantini, A.; Degraeve, R.; Raghavan, N.; Nigon, R.; Strangio, S.; Kar, G.; Wouters, D.J.; Chen, Y.Y.; Komura, M.; et al. Understanding of the intrinsic characteristics and memory trade-offs of sub-μA filamentary RRAM operation. In Proceedings of the 2013 Symposium on VLSI Technology, Kyoto, Japan, 11–14 June 2013.
10. Gonzalez, M.B.; Rafí, J.M.; Beldarrain, O.; Zabala, M.; Campabadal, F. Analysis of the Switching Variability in Ni/HfO_2-based RRAM Devices. *IEEE Trans. Device Mater. Reliab.* **2014**, *14*, 769–771.
11. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. In Proceedings of the Neural Information Processing Systems, 2012, 1097-1105.
12. Goodfellow, I.; Yoshua, B.; Aaron, C. *Deep Learning*; MIT press: Cambridge, MA, USA, 2016.
13. Ielmini, D.; Rainer, W. *Resistive Switching: From Fundamentals of Nanoionic Redox Processes to Memristive Device Applications*; John Wiley & Sons: Hoboken, NJ, USA, 2015.
14. Wang, I.T.; Lin, Y.C.; Wang, Y.F.; Hsu, C.W.; Hou, T.H. 3D synaptic architecture with ultralow sub-10 fJ energy per spike for neuromorphic computation. In Proceedings of the 2014 IEEE International Electron Devices Meeting (IEDM), San Francisco, CA, USA, 13–17 December 2014.
15. Qiu, K.; Chen, W.; Xu, Y.; Xia, L.; Wang, Y.; Shao, Z. A peripheral circuit reuse structure integrated with a retimed data flow for low power RRAM crossbar-based CNN. In Proceedings of the 2018 Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden, Germany, 19–23 March 2018.

16. Xiang, Ya. Impacts of State Instability and Retention Failure of Filamentary Analog RRAM on the Performance of Deep Neural Network. *IEEE Trans. Electron Devices* **2019**, *66*, 4517–4522.
17. Zhao, M. Characterizing endurance degradation of incremental switching in analog RRAM for neuromorphic systems. In Proceedings of the 2018 IEEE International Electron Devices Meeting (IEDM), San Francisco, CA, USA, 1–5 December 2018.
18. Woo, J.; Yu, S. Impact of Selector Devices in Analog RRAM-Based Crossbar Arrays for Inference and Training of Neuromorphic System. *IEEE Trans. Very Large Scale Integr. VLSI Syst*. **2019**, *27*, 2205–2212.