frontiers | Frontiers in Sensors

ORIGINAL RESEARCH
published: 13 May 2022
doi: 10.3389/fsens.2022.896299

# Fault-Aware Adversary Attack Analyses and Enhancement for RRAM-Based Neuromorphic Accelerator

*Liuting Shang*[1]*, *Sungyong Jung*[1], *Fengjun Li*[2] and *Chenyun Pan*[1]

[1]Department of Electrical Engineering, University of Texas at Arlington, Arlington, TX, United States, [2]Department of Electrical Engineering and Computer Science, University of Kansas, Lawrence, KS, United States

Neural networks have been widely deployed in sensor networks and IoT systems due to the advance in lightweight design and edge computing as well as emerging energy-efficient neuromorphic accelerators. However, adversary attack has raised a major threat against neural networks, which can be further enhanced by leveraging the natural hard faults in the neuromorphic accelerator that is based on resistive random access memory (RRAM). In this paper, we perform a comprehensive fault-aware attack analysis method for RRAM-based accelerators by considering five attack models based on a wide range of device- and circuit-level nonideal properties. The research on nonideal properties takes into account detailed hardware situations and provides a more accurate perspective on security. Compared to the existing adversary attack strategy that only leverages the natural fault, we propose an initiative attack based on two soft fault injection methods, which do not require a high-precision laboratory environment. In addition, an optimized fault-aware adversary algorithm is also proposed to enhance the attack effectiveness. The simulation results of an MNIST dataset on a classic convolutional neural network have shown that the proposed fault-aware adversary attack models and algorithms achieve a significant improvement in the attacking image classification.

**Keywords: security of IoT systems, hardware security, adversary attack, neuromorphic accelerator, nonideal property, fault injection**

OPEN ACCESS

***Edited by:***
*Giovanni Pau,*
*Kore University of Enna, Italy*

***Reviewed by:***
*Xing Wu,*
*East China Normal University, China*
*Yuanfang Chen,*
*Hangzhou Dianzi University, China*
*Bing Li,*
*Capital Normal University, China*

***Correspondence:**
*Liuting Shang*
*liuting.shang@mavs.uta.edu*

***Specialty section:***
*This article was submitted to Sensor Networks, a section of the journal Frontiers in Sensors*

***Received:*** *14 March 2022*
***Accepted:*** *26 April 2022*
***Published:*** *13 May 2022*

***Citation:***
*Shang L, Jung S, Li F and Pan C (2022) Fault-Aware Adversary Attack Analyses and Enhancement for RRAM-Based Neuromorphic Accelerator. Front. Sens. 3:896299. doi: 10.3389/fsens.2022.896299*

## INTRODUCTION

The rapid development of deep learning algorithms and hardware in recent years has brought great success in a wide range of applications, some of which serve distributed systems such as the Internet of Things (IoT) and computer vision. However, deep learning meets two aspects of hardware problems in real-world implementation, especially in distributed scenarios. On the one hand, it is a severe challenge to meet various requirements of neural applications for the current hardware, including but not limited to computation throughput, latency, energy efficiency, and bandwidth efficiency in both training and inference stages. As a consequence, many accelerators schemes have been explored to better support the deployment of the neural network, in which the RRAM-based neuromorphic circuit is one of the most promising schemes that provides orders of magnitude improvement in the area, energy, and speed compared to

CMOS-based platform/accelerators, such as TPU/GPU (Kim et al., 2012; Giacomin et al., 2018; Hu et al., 2018; Liu et al., 2018). On the other hand, privacy leakage is a critical threat in distributed systems from a security perspective. Sending data from the sensor networks to the cloud can incur the users' concern that their secure data/behavioral information will be illegally captured by malicious agencies/persons (Chen and Ran, 2019). Hence, edge computing that deploys a computation engine in-site and locally processes the data captured from sensors becomes an attractive option for applications in distributed systems (AWS, 2022; Ha et al., 2014; Zhang et al., 2015; Hung et al., 2018; Mohammadi et al., 2018; Chinchali et al., 2018; Liu et al., 2019). Meanwhile, a distributed system can benefit from both RRAM-based accelerator and edge computing to achieve better performance, privacy, and longer working life (Hsu et al., 2019; Zhou et al., 2019; Singh et al., 2021).

Unfortunately, the trend of deploying RRAM-based accelerators in edge computing raises new security risks. Traditionally, neural networks are vulnerable to well-designed adversary attacks, which misleads classification by adding human-unnoticeable perturbation on the input samples. For example, adversary attacks can efficiently corrupt the intelligence of image classification or even manipulate the result of classification by slightly modifying the pixels in an image (Goodfellow et al., 2014; Carlini and Wagner, 2017; Madry et al., 2017). Although many software-domain defense strategies, such as adversarial training (Goodfellow et al., 2014), gradients masking (Papernot et al., 2017), and model distillation (Papernot et al., 2016), have been developed to substantially reduce the adversary attack success rate, hardware-based adversary attacks are proposed to further enhance the attack. Because of the distributed deployment of the RRAM-based accelerator, edge computing hardware is accessible to adversaries. Several hardware-based adversary attack methods have been designed to largely strengthen misleading ability by actively injecting faults to the weights of neural networks in digital memory. The fault injection methods target bit-flipping on critical positions by remote trojans, such as row hammer attacks (Rakin et al., 2020; Rakin et al., 2021), or invasive physical attacks like the laser injection technique (Liu et al., 2017a; Breier et al., 2018). However, those methods mainly target the digital system, and the security investigations for emerging RRAM-based analog accelerators are still insufficient. As emerging devices, RRAMs usually suffer from the immaturity of the fabrication technology and exhibit natural hard/soft faults (resistance stuck or drift), especially those that have multiple resistance states and represent multi-bit weights in a single cell. Since the neuromorphic system possesses an inherent error tolerance, minor nonideal characteristics in devices cannot induce noticeable accuracy degradation and can be considered "benign" (Temam, 2012). The RRAM-based neural network accelerators with 'benign' faults can properly operate pre-trained functions in the testing/operation but are vulnerable to fault-aware adversary attacks.

A few works have investigated the impact of 'benign' nonideal properties in RRAM-based hardware on adversary attacks toward image classification. One work discusses how the nonideal properties reduce the adversary attack success rate and concludes that the RRAM-based neuromorphic hardware is inherently robust against adversary attacks (Bhattacharjee and Panda, 2020). While a recent work points out that the hard faults in the RRAM crossbar array can be leveraged to substantially enhance the adversary attack strength and effectively breakthrough software defense strategy (Lv et al., 2021). In this previous work, the investigation is focused on the software domain and the nonideal properties are simplified as 'hard fault', i.e., the corrupted weights are fixed to the maximum or minimum value of a layer in neural networks. In this paper, we will analyze and enhance the faults/variation-based adversary attack based on rich RRAM nonideal behaviors and circuit characteristics in neural network accelerators. The major contributions of this work are highlighted in the following.

- This work develops adversary attack models based on the rich and detailed nonideal properties that exist in the RRAM crossbar array, including soft-faults (i.e., conductance variation, hereinafter referred to as 'variation') and hard-faults. A novel perspective of nonideal properties at the circuit level is provided, including the realization of signed weights and the distributions of faults considering the mapping strategy. Such a perspective promotes the scope of hardware-aware adversary attacks.
- The device-level and circuit-level attack models investigated in this paper enable the active enhancing/creating of the natural faults/variation and increase the effectiveness of attacks. By using normal images, projected gradient descent (PGD) adversary images, and fault-aware adversary images as inputs, comprehensive evaluations of performance are performed with a convolutional neural network trained with the MNIST handwritten digits dataset.
- We enhance the fault-aware adversary attack method by reducing the amplitude of perturbation and increasing the number of perturbed pixels in input images. Without increasing the total change in pixels values, the enhanced algorithm not only improves the ability to mislead the classifier but also eliminates the obvious traces of changes.

The rest of this work is organized as follows. In *Introduction*, the background of conventional and fault-aware adversary attacks is provided. In *Introduction*, the properties of emerging RRAM devices and RRAM-based neuromorphic circuits that can be leveraged by adversary attacks are introduced. The experiment setup as well as the proposed fault-aware adversary attack models are given in *Introduction*. We provide the simulation results of the attack models and analyze them in *Introduction*. In *Introduction*, an enhanced attack algorithm is proposed and evaluated. Finally, we discuss several potential defense methods in *Introduction*.

## BACKGROUND OF CONVENTIONAL AND FAULT-AWARE ADVERSARY ATTACK

Researchers have developed rich adversary attack techniques to undermine the security of neural networks in various areas. Here,

we discuss one of the most popular attack targets, neural networks for image classification. The purpose of adversary attack can be concluded as adding human-invisible perturbations in test images to mislead the classification, which can be expressed as:

$$\boldsymbol{C}\left(\boldsymbol{W}, \boldsymbol{x_i} + \boldsymbol{\delta_p}\right) \neq \boldsymbol{C}\left(\boldsymbol{W}, \boldsymbol{x_i}\right), \ \left\|\boldsymbol{\delta_{p_n}}\right\| < \epsilon \tag{1}$$

where C (•) refers to the predictions of the neural network, $W$, $x_i$, and $\delta_p$, denotes the network weights, test image, and perturbations, respectively, and $\epsilon$ represents the constraints of perturbations and ensures that the image edition cannot be easily recognized by a human. Conventional adversary attack methods usually add the perturbation test images by using the gradients (Goodfellow et al., 2014; Carlini and Wagner, 2017; Madry et al., 2017), while defense techniques are proposed as effective countermeasures (Goodfellow et al., 2014; Papernot et al., 2016; Papernot et al., 2017).

In a previous work (Lv et al., 2021), natural hard-fault in RRAM is utilized to break through protected neural network accelerators. Such an attack bypasses conventional protection by assuming users will ignore the benign hard fault in hardware. Compared to the fault-aware attack that injects faults by Trojan/physical methods, it is harder to prevent this method because the faults utilized are naturally existing. The fault-aware adversary attack on an RRAM-based accelerator first measures the faults in the RRAMs that store the weights of a neural network. Then, attackers find perturbation-vulnerable positions in a given test image $x_i$. The standard of vulnerability is the gradient difference $g_a$, which is defined by (Lv et al., 2021):

$$\boldsymbol{g_a}\left(\boldsymbol{x_i}\right) = \boldsymbol{g'}\left(\boldsymbol{x_i}\right) - \boldsymbol{g}\left(\boldsymbol{x_i}\right) \tag{2}$$

where gradient $g'$ of test image $x_i$ is calculated by a neural network model with fault, and gradient $g$ is calculated in the fault-free model. For a pixel with coordinator $(j,k)$, a larger $g_a(x_i(j,k))$ indicates a stronger ability to distort the prediction, so the fault-aware adversary attack selects the pixel with the largest $g_a$ to add perturbation. Starting from a clean image, the perturbation is added by following a greedy algorithm, i.e., gradually increasing the number of perturbed pixels and the amplitude of the newly added perturbation until the classification is mislead or the perturbation reaches the limitation. We adopt this algorithm to test the proposed attack models for different fault types and design an enhanced method based on it.

# ATTACKABLE DEVICE/CIRCUIT PROPERTIES

In this section, we investigate the properties of emerging RRAM devices and RRAM-based neuromorphic circuits, which build the foundation of fault-aware adversary attack models that are proposed in *Attackable Device/Circuit Properties*.

## Hard/Soft Faults in RRAM Devices

RRAM devices are area-competitive (multi-level data in one $4F^2$ cell (Münch et al., 2019)) for large-scale neuromorphic circuits partly because of their compact nanoscale integration. However, this feature also leads to lower reliability, i.e., more hard/soft faults. There are multiple mechanism candidates for building RRAM cells, ranging from filamentary (Prezioso et al., 2015; Yu et al., 2015), phase change (Burr et al., 2015; Kim et al., 2015), ferroelectric ram (Kaneko et al., 2014), to conductive-bridge (Jo et al., 2010; Jeong et al., 2015). For most device mechanisms, the realization and retention of multi-bit data storage on the nanoscale are very challenging due to several issues.

First, permanently hard faults (i.e., stuck at On/Off resistance state) are common due to improper read/write current/voltage, environment stimulus, and/or natural damage (Lv et al., 2015; Jiang et al., 2016; He et al., 2019; Yeo et al., 2019; Fadeev and Rudenko, 2021; Wu et al., 2021). Second, imperfections exist in the complicated fabrication process, such as feature size shrinkage, line-edge roughness, and oxide thickness fluctuation can lead to variations in the resistance of RRAM (i.e., soft faults). Although minor variations can be compensated during the programming process of RRAM, some faults that lead to extremely high or low conductance cannot be fixed once the fabrication is done. In addition, because of the inherent particle drift and the currents periodically passthrough RRAM during the operation, the values stored in RRAM vary with time (Lv et al., 2015; Jiang et al., 2016). Without rewriting weight values, the accuracy of stored weights in RRAMs will degrade even if the chip is not operating (Fadeev and Rudenko, 2021). Moreover, all environmental factors, such as temperature, electrical fields, magnetic fields, and illumination, affect the resistance state. For example, the temperature has been shown to have a strong positive correlation with the conductance variation/fault rate of RRAM (Lv et al., 2015; Jiang et al., 2016; Fadeev and Rudenko, 2021; Wu et al., 2021). Such influence of temperature may be exaggerated by the chip cooling limitation in a dense layout, such as 3-dimensional stacked RRAM architecture for higher area efficiency (An et al., 2019; Lin et al., 2020). In addition, the distribution of variation/fault rate changes with the uneven distribution of corresponding factors. For example, the center positions in 3-dimensional stacking RRAM architecture encounter more faults due to a higher temperature (Beigi and Memik, 2016), and the locally enhanced electric field generated by the irregular shapes at the edge of the filamentary RRAM array causes more faults (Lv et al., 2015). All the sources of soft/hard faults above can be utilized by adversaries to mislead the neural network.

There are several types of solutions to reduce the variation mentioned above. For permanent variation/fault, solutions at the programing stage have been proposed to alleviate the hard faults and soft faults caused by inherent mismatches (Tunali and Altun, 2016; Liu et al., 2017b; Xia et al., 2017; Xia et al., 2018). However, some solutions (Liu et al., 2017b; Xia et al., 2017) require redundant hardware to replace corrupted memristors with the functional ones, and other solutions (Tunali and Altun, 2016; Liu et al., 2017b; Xia et al., 2018) need error correction using additional memristors (Tunali and Altun, 2016; Liu et al., 2017b; Xia et al., 2018). The additional memristors and error-correcting circuits significantly increase the costs and create

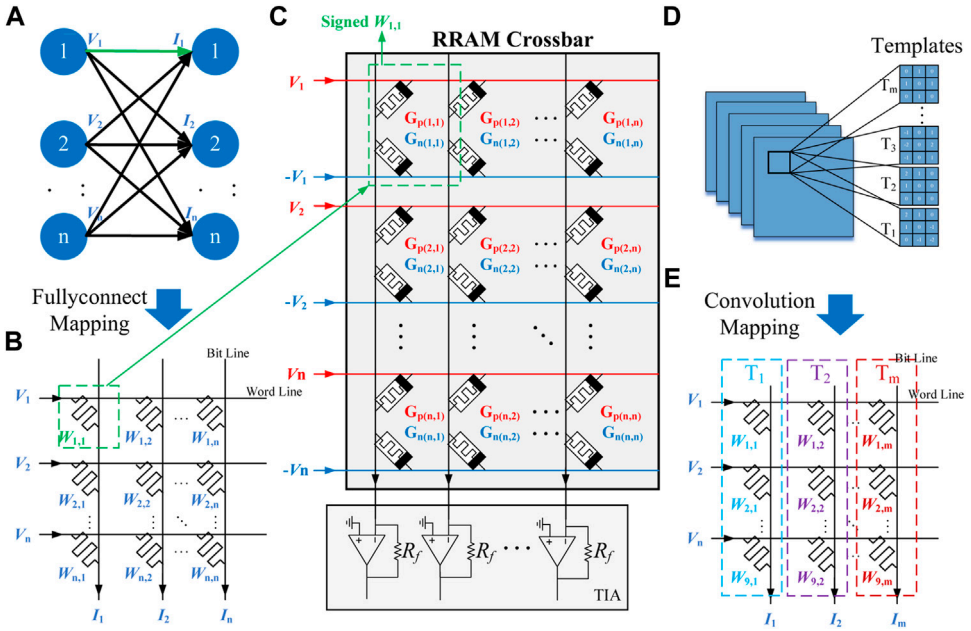


**FIGURE 1 | (A)** Diagram of fully-connected layer, **(B)** assumed 1-RRAM-1-weight weights mapping in previous work, **(C)** mapping signed weights by using complementary RRAMs to represent positive and negative values, **(D)** diagram of convolution layer, and **(E)** mapping method for convolution layer.

difficulty in hardware implementation as the depth of the neural network grows. For temporary variation/fault, a straightforward solution is to simply check and rewrite all the weights stored in RRAMs during the inference period (Xia et al., 2018), which is time- and energy-consuming due to the complicated process, a large number of weights, and requirement of data communication. Since IoT/sensor networks are energy-sensitive, such a fault detection process cannot be frequently performed when the classification accuracy of neural networks has not met a noticeable degradation. In addition, adding cooling equipment, protecting mask, better quality control, and specific design can alleviate the temporary variation/fault. However, prevention of temporary variation/fault may induce a significant challenge and cost because in many cases sensors/computation nodes in distributed network may work in a variety of unstable outdoor harsh environments, and these devices are considered as low-cost consumables. In conclusion, the benign faults will widely exist with the neuromorphic accelerator for a long period of time.

## RRAM Crossbar Array and Circuit Design

RRAM crossbar array is the core part of a neural network accelerator, which performs the most computation-dense MVM operation. In this section, we analyze the circuit-level model of the RRAM array and the corresponding impact of variation/fault on the neural network.

The first noticeable point is the representation strategy of weights. As shown in **Figure 1B**, MVM (V • W) is realized by programming the conductance of the RRAM cells to store the weight matrix (W). Then, the analog voltages are applied to the word line to represent the input vector (V). The currents flowing through RRAM cells are accumulated in bit lines and transformed to voltages by trans-impedance amplifiers (TIA) for the multiplication/summation results. In the previous work, the weight is represented by a single cell as shown in **Figure 1B**, and it assumes the faulty weights are stuck at the maximum/minimum value among cells in a neural layer. While according to a recent state-of-the-art work that achieves a real-world neuromorphic accelerator (Yao et al., 2020), using two memristors to represent one weight value is more practical and precise and leads to easier implementation. As shown in **Figure 1C**, $W_{1,1}$ is represented by the conductance of a pair of RRAM cells, $G_{p(1,1)}$ and $G_{n(1,1)}$, which are connected to input $V_1$ and inversed input $-V_1$, respectively. If $W_{1,1}$ is a negative value, the current flowing into the bit line can be calculated as:

$$I_{(1,1)} = V_1\left(G_{p(1,1)} - G_{n(1,1)}\right),\ G_{p(1,1)} \ll G_{n(1,1)} \tag{3}$$

Since the minimum conductance value is usually not within the stable working conductance range for storing data, the RRAM cell that does not store information ($G_{p(1,1)}$ in **Eq 3**) can be regarded as negligible.

If one of the RRAM cells in a pair is faulty and stuck at maximum or minimum conductance, seven possible

**TABLE 1 |** All eight results of getting 1 cell stuck at minimum/maximum conductance.

| | $G_p$ -> $G_{min}$ | $G_n$ -> $G_{min}$ | $G_p$ -> $G_{max}$ | $G_n$ -> $G_{max}$ |
|---|---|---|---|---|
| $+\|W_{(i,j)}\|$ | 0 | $+\|W_{(i,j)}\|$ | $+\|W_{max}\|$ | $\|W_{(i,j)}\|$-$\|W_{max}\|$ |
| $-\|W_{(i,j)}\|$ | $-\|W_{(i,j)}\|$ | 0 | $\|W_{max}\|$-$\|W_{(i,j)}\|$ | $-\|W_{max}\|$ |

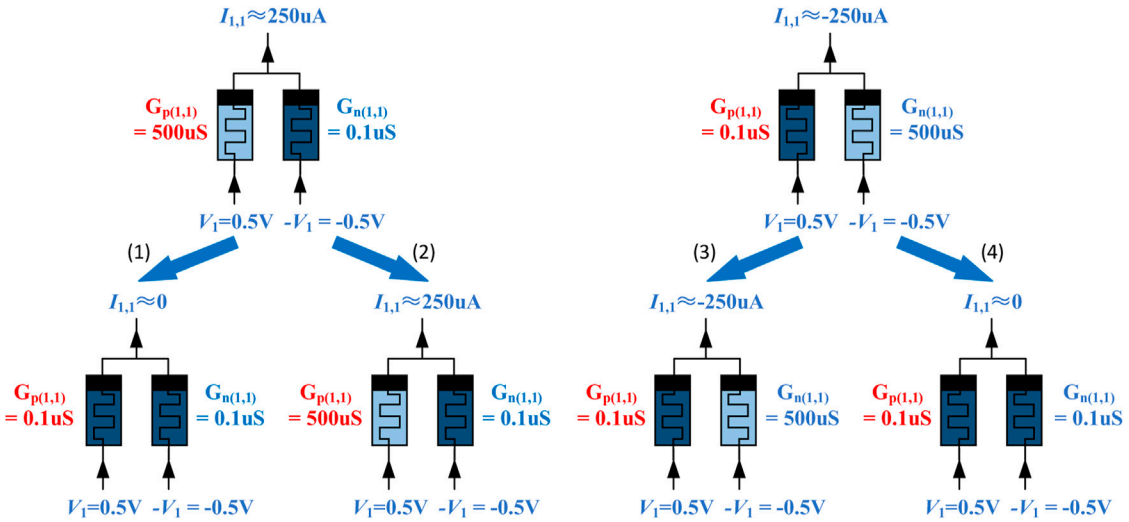


**FIGURE 2 |** Circuit diagram of four results of getting 1 cell stuck at minimum conductance.

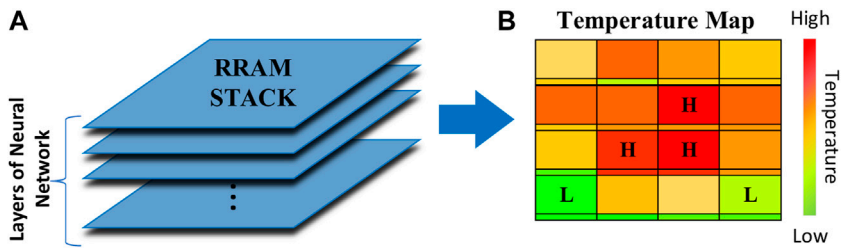


**FIGURE 3 | (A)** 3-dimensional RRAM crossbar array stack for higher area efficiency, **(B)** temperature map in RRAM stack.

consequences can happen as shown in **Table 1**; **Figure 2** illustrates how a weight value is changed when one of the RRAM cells in the pair is stuck at minimum conductance. It can be observed that directly using the device fault rate for the evaluation of fault-aware adversary attack rate causes an overestimation in the weight difference.

Second, the method of mapping weights to the RRAM crossbar array affects how the variation and fault distribute in weights. As discussed above, multiple reasons can lead to the uneven distribution of variation or faults. As an example, the temperature model in **Figures 3A, B** shows that variations/faults have a higher probability to occur in the center area of each layer of the RRAM stack because of higher temperatures. By assuming one layer contains one crossbar array, the mapping-aware variation/fault analyses can be performed to investigate how different fault-dense areas affect the adversary attack. In this paper, we consider the mapping scheme of the two most common layers in neural networks, i.e., the fully-connected layer and convolution layer. A mapping method of the fully-connected layer can be found in **Figures 1A, B**, where input $V_i$ is the output of the upper layer, and $I_i$ is the input of the lower layer. Hence, $W_{2,3}$ denotes the weight connecting nodes one to two and 2–3. The mapping method of the convolution layer can be found in **Figures 1D, E**, where templates are reshaped as vectors and placed in the bit line direction in the crossbar array. Since variation and fault is more likely to happen in the center area, they will affect different parts of weights in different layers. The affection and analysis will be investigated in *RRAM Crossbar Array and Circuit Design*.

Fourth, due to the immature fabrication process of emerging RRAM devices, a special precision extension technique can be deployed to improve the weight precision. For example, **Figure 4** shows a method of using three 2-bit RRAM cells to represent one 6-bit weight value by applying quantified input voltages and summing three output currents. This exposes the accelerator to a novel risk of attack towards quantification.

## NOVEL ATTACK MODELS

According to the device properties and the details of circuit design investigated in *Novel Attack Models*, we propose five fault-aware attack models. Several assumptions are first given,

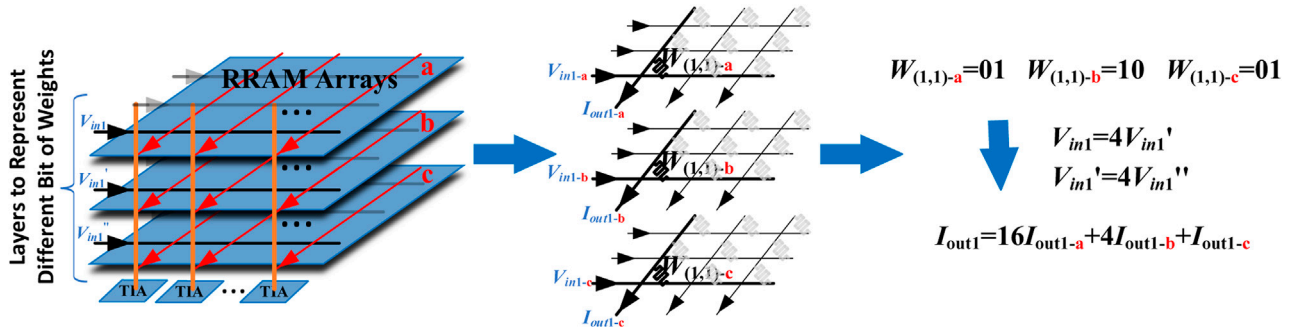


**FIGURE 4 |** Diagram of using multiple RRAMs to extend the precision of one weight value. Here, three 2-bit cells are utilized to represent a weight value of 6-bit precision, and the input voltages are scaled to different weights of each cell.

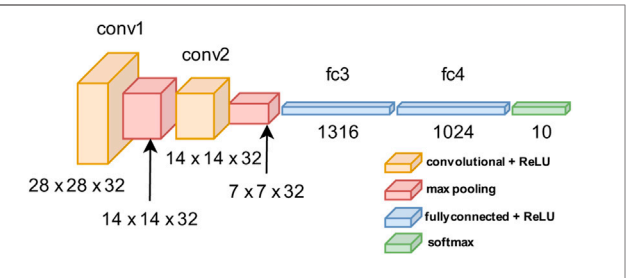


**FIGURE 5 |** Architecture of neural network for MNIST.

then modelized attack strategies are designed to prepare for further benchmarking.

## Assumptions of Adversary Attack

We adopt the same assumptions in previous work for investigating the fault-aware adversary attack (Lv et al., 2021):

(1) White box: the adversary has access to the full architecture and parameters of the neural network model, as well as the labels for provided inputs.
(2) Model of neural networks: the details of neural network architecture are provided in **Figure 5**, in which convolution layers and fully-connected layers are followed by the ReLU activation function. We also deploy an existing adversarial training defense mechanism to strengthen the robustness of neural networks against adversarial attacks (Madry et al., 2017).
(3) Data set and perturbations: the fault-aware adversary examples are generated by adding extra perturbations to the conventional adversarial examples (M. A. E. Challenge, 2017). The conventional adversarial examples have $l_\infty$ norm of perturbations on pixels that do not exceed $\epsilon = 0.3$ in **Eq 1**. For the extra fault-targeted perturbations, the restriction of their amplitude is the final pixel value should not exceed [0, 1]. To maintain the concealment of perturbation against the human eye, only around 1% (10 pixels for MNIST data set) of fault-targeted perturbations are allowed. The attack framework is built on Tensorflow.

## Attack Models

According to the discussion in *Attack Models*, an original binary attack model and five novel variation/fault attack models are developed as shown in **Figure 6**. By independently analyzing each of them, the attack models can be accordingly categorized as follows.

Original Baseline Binary Attack Model: this model adopts the same configurations in previous work (Lv et al., 2021), which assumes all the faults are evenly distributed, faulty values are positive or negative maximum absolute values (i.e., binary) in a layer. Meanwhile, all the parameters, as well as faulty values in the neural network, are known to the attacker.

Attack Model 1: regarding the fault model in **Figure 6A**, we assume only 1 cell in a pair is faulty because of the low fault rate. Thus, the random hard faults in the RRAM crossbar array result in seven kinds of change in each weight value according to **Table 1**. To verify the necessity of this model, we also investigate how the attack can be affected if adversaries simplify Attack Model 1, i.e., apply the Baseline Attack Model to circuits with a dual RRAM representation. The faults could be the natural faults during the fabrication, or the faults created during a long operation. Because the adversaries have access to the neural network accelerator, fault testing algorithms, such as March-C, can be applied to detect the hardware faults information (Chen et al., 2014; Liu et al., 2016). With unlimited access to the peripheral read/write circuits in a non-invasive way, attackers can also read out the conduction of the functional cell in the faulty pair and calculate the real stored weight value. Notably, based on the relation between temperature and fault rate, adversaries can actively apply long-term temperature raising or short-term baking to increase the fault rate in the RRAM chip. Compared to existing laser-based fault injection that is accurate to bit, the temperature-based approach is inexpensive and can be performed *in-situ* instead of in-laboratory environments. That also makes the attack fast and stealthy without the need for the device to be offline. Furthermore, the temperature-based active fault injection reduces the inherent faults that are vulnerable to the adversary attack.

Attack Model 2: as shown in **Figure 6B**, for the RRAM arrays that do not have enough hard faults to trigger the fault-based

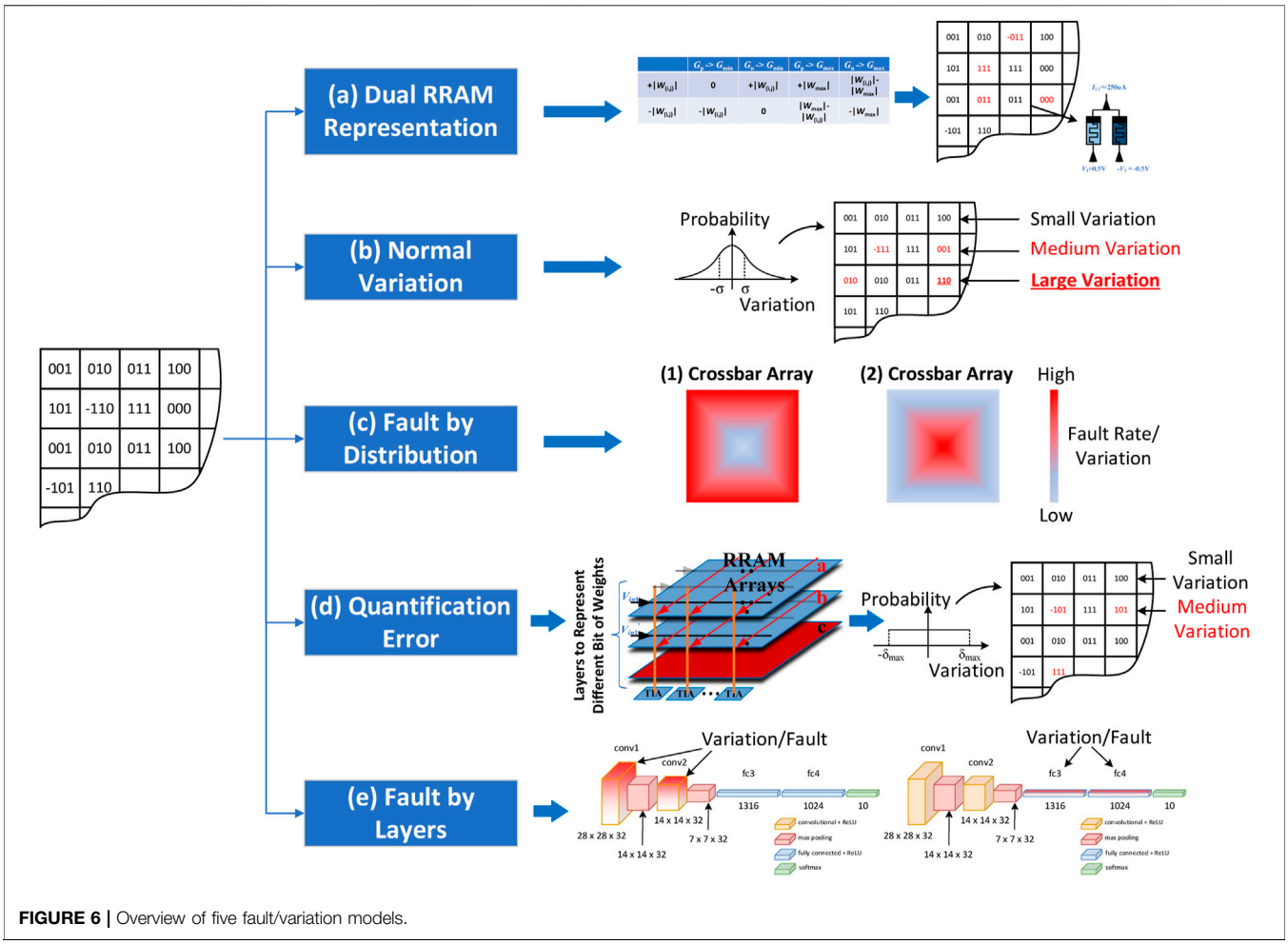


**FIGURE 6 |** Overview of five fault/variation models.

adversary attack, adversaries can utilize the variations in the RRAM chip to enhance the attack ability. As discussed in *Attack Models*, the variation in conductance widely exists in hardware and deteriorates with the increase in operating time and temperature. Adversaries are able to access the variation information by leveraging the inherent read function, which is slower than fault detection because of the large number of parameters in deep neural networks. However, it is still advantageous for convenient *in-situ* attacks and the ability to enhance variations by manipulating operation temperature. The amplitude of variation in every RRAM cell obeys a normal distribution with the standard deviation $\sigma$ (Chen et al., 2017).

Attack Model 3: for the RRAM array that utilizes 3-dimensional RRAM stacks or those that have more defects in fabrication, adversaries can leverage the higher fault rate caused by thermal or other stimuli. While the positions of such faults in an array are unevenly distributed according to the type of the stimuli. Hence, using the thermal and electric field fault mechanisms discussed in *Attack Models* as examples, we create the models of the distribution of fault positions as shown in **Figure 6C** to investigate the impact of fault distributions on the attack success rate. The fault type in this model is assumed to be hard faults. The fault rates of RRAM cells in the first model (c-1) and the non-fault rate in the second model (c-2) are assumed to be exponentially related to the distance from the cell to the center of the array. For example, the fault rate in the model (c-1) can be expressed as:

$$P_f(i,j) = 2^{\sqrt{\left(\left(i-\frac{M}{2}\right)^2+\left(j-\frac{N}{2}\right)^2\right)}} \tag{4}$$

where $(i, j)$ is an arbitrary position in an $M \times N$ array. Also, the two uneven distributions are normalized to ensure the same total fault rate as the baseline binary attack model.

Attack Model 4: for the weight representation method that extends the precision by using more RRAM devices (i.e., more MVM blocks), we propose the attack model as shown in **Figure 6D**. Since the improved weight precision in the RRAM-based neural accelerator comes with the price of more crossbar arrays as well as peripheral circuits, power gating schemes can be deployed to provide a customizable precision. The energy can be optimized accordingly by cutting off the power

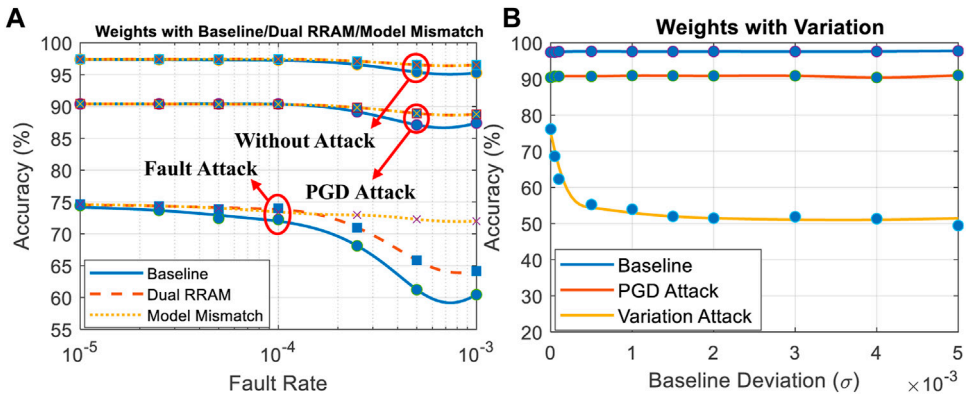


**FIGURE 7 | (A)** Results of Attack Model 1, and **(B)** results of Attack Model 2.

supply for the MVM block that represents the least significant bits. However, adversaries can access the hardware and maliciously shut down those MVM blocks to create a quantification error, which intentionally induces the conductance variation within $[-\delta, 0]$ when rounding toward the nearest representable number, where $\delta$ is the minimum resolution of the degraded weight value.

Attack Model 5: the attack model is shown in **Figure 6E** aims to investigate the sensitivity of fault-aware adversary attacks regarding the positions and types of layers. To avoid reading all parameters and accelerating the attack process, adversaries can only enhance and detect the faults in layers that are most effective in improving the attack success rate. Here, this attack model is separated by two assumptions. The first one assumes that the original accelerator does not have a fault and randomly creates faults to the layers that are effective to adversaries (for example, by locally increasing temperature). Another is a control group that only reads part of layers from an all-layer faulty chip, which aims to test whether the absence of partial fault information can disable the attack. For both assumptions, we set the convolution layers (close to input) or fully-connected layers (close to output) as the layers that we acknowledge all the faults as shown in **Figure 6E**.

# EVALUATION OF THE PROPOSED ATTACK MODELS

## Configuration of Evaluation

In order to obtain a robust adversary attack model, first, the given neural network is fully trained by the existing method (Madry et al., 2017; M. A. E. Challenge). Then, the faults or variations are added to simulate the potential and real conditions of the hardware. Since all the discussed faults are due to the nonideality of the RRAM devices in a crossbar array, the faults/variation are only added to the weights among layers, while the existing work adds faults to all trainable variables, including bias (Lv et al., 2021). This setting and the training method will lead to a difference in the accuracy and attack success rate. Here, the attack success rate is defined as the baseline accuracy minus accuracy after the attack. We use 500 MNIST handwriting figures as testing examples to create conventional adversary examples as well as fault-aware adversary examples. For each fault/variation setting, neural network models with 20 different faults/variation are generated for the purpose of sampling.

## Evaluation Results

The simulation results from Attack Model one to five are shown in the figures below. 'Without Attack', 'PGD', and 'Faults/Variation Attack' denote inputting image examples with no perturbation, PGD adversary perturbation, and faults/variation-based adversary perturbation, respectively. The curves for the baseline prediction show that the RRAM nonideality in neuromorphic accelerator does not lead to noticeable degradation in figure classification. Meanwhile, the lines of the PGD attack indicate that the conventional adversary attack cannot efficiently leverage the natural faults/variation in hardware to enhance the attack success rate.

**Figure 7A** includes the attack success rates generated by the Binary Attack Model and Attack Model 1. The result shows that Attack Model one results in a higher classification rate, which is because two out of the seven conditions in **Table 1** will not lead to an error in weight value. The dotted curve shows that applying Baseline Attack Model to a circuit with dual RRAM representation will lead to degradation in attack effectiveness. This result indicates that it is necessary to select a correct attack model to ensure the attack success rate.

The results of Attack Model two are shown in **Figure 7B**, in which the variation-based adversary attack effectively reduces the worst accuracy in the baseline attack model from 60.5 to 49.4%. That means the widely existing minor variation in RRAM crossbar arrays can be utilized to create a larger impact on the final predicted results. the hard faults can be quickly detected by array-level scanning and replaced by backup/spare cells, the variation in RRAM possesses better concealment since comparing the conductance of each RRAM cell with standard value is extremely time-consuming.

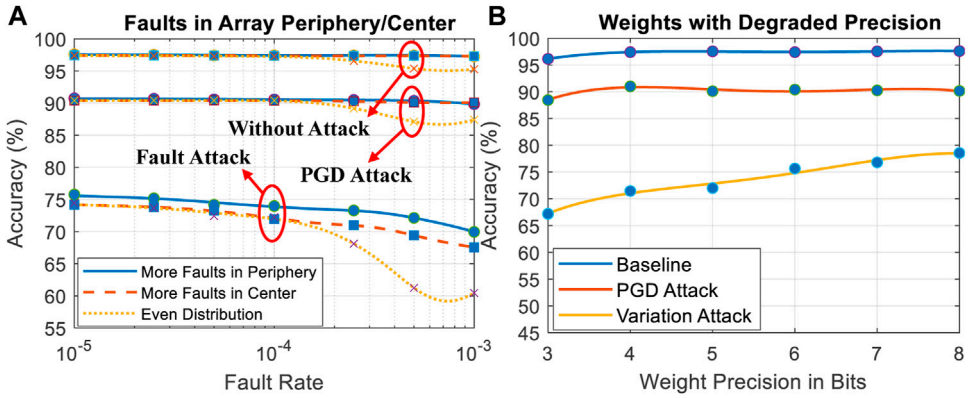


**FIGURE 8 | (A)** Results of attack Model 3, and **(B)** results of attack Model 4.

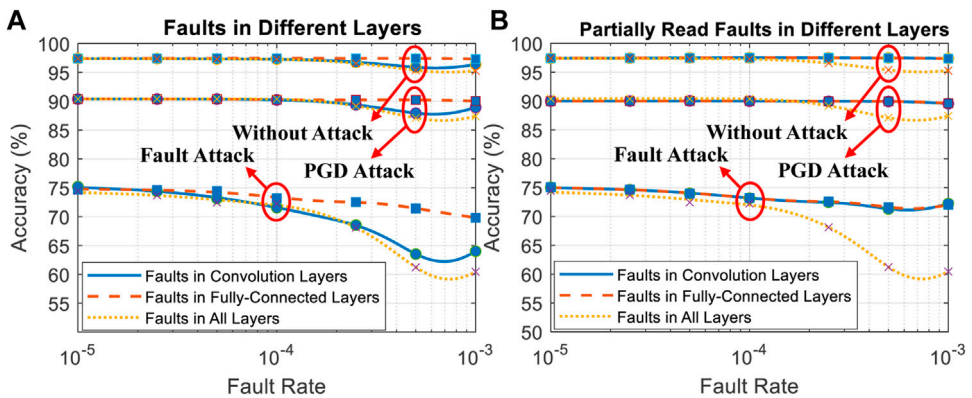


**FIGURE 9 | (A)** Results of attack Model 5, in which all faults in the chip are known, **(B)** results of attack Model 5, in which only the faults in certain layers are known.

In **Figure 8A**, we can observe that two uneven distributions of faults in an RRAM crossbar array, i.e., Attack Model 3, reduce the attack success rate. This indicates that the attack is less effective if the positions of faults are dominated by unevenly distributed stimuli or fabrication mismatch.

The attack results of Attack Model four are shown in **Figure 8B**, which claims precision degradation as a promising method to create value variation for adversary attacks. The attack effectiveness proven by Attack Model two and four indicates that not only hard fault but also the small conductance variation can lead to significant gradients difference and open a door for adversary attack.

In **Figure 9A**, adding faults in convolution layers, i.e., the first two layers, results in a much lower accuracy compared to the attacked classifier that only has faults in fully-connected layers. Although fully-connected layers have a dominant number of parameters (99.06%) in the neural network used for simulation, those layers do not play an important role in the fault-based attack. On the contrary, only adding faults to convolution layers achieves comparable attack effectiveness compared to the Original Binary Attack Model, which has faults in all layers. This proves the hypothesis in previous work, i.e., the faults in neural accelerator need several layers to broadcast and thus enhance the impact on classification.

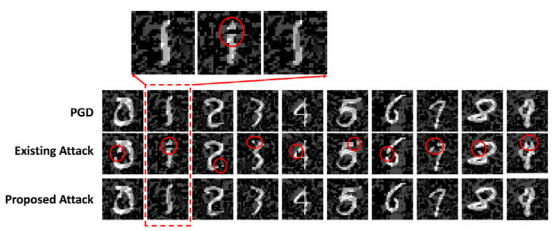


**FIGURE 10 |** Comparison of adversary attack images on MNIST. The disconnected positions are marked by red circles. The images generated by the proposed attack is with perturbations limited within [-0.2, 0.2].

However, **Figure 9B** indicates that only detecting partial faults of an accelerator that has faults in every layer will weaken the fault-aware adversary attack. provide a conclusion that the *in-situ* faults detection can obtain ~100 × acceleration if the active fault injection can be focused on the area that is closer to the input (for

**TABLE 2 |** All eight results of getting 1 cell stuck at minimum/maximum conductance. 10 perturbations occur in the range of [-1.1] and the remaining ranges have 20 perturbations.

| Rang of perturbation | [-1.1] | [-1.1] or [-0.35.0.35] | Traverse 5 ranges | [-0.1.0.1] | [-0.2.0.2] | [-0.3.0.3] | [-0.4.0.4] | [-0.5.0.5] | [-0.6.0.6] | [-0.7.0.7] |
|---|---|---|---|---|---|---|---|---|---|---|
| Accuracy | 60.5% | 43.3% | 40.6% | 77.9% | 69.4% | 57.7% | 50.8% | 45.1% | 40.8% | 40.7% |
| Average Changed Pixel Values | 4.1 | 4.1 | 4.0 | 0.87 | 1.0 | 2.8 | 3.2 | 4.6 | 5.6 | 6.4 |

**TABLE 3 |** Comparison of original and enhanced adversary attacks on hard faults (dual RRAM) and soft faults (normal variation and degraded precision). The five ranges for the enhanced attack are the same as in **Table 2**.

| Other fault types | | Dual RRAM | Variation ($\sigma = 5 \times 10^{-3}$) | Precision (bit = 3) |
|---|---|---|---|---|
| Original Attack | Accuracy | 64.1% | 46.4% | 67.2% |
| | Average Changed Pixel Values | 4.1 | 4.0 | 4.2 |
| Enhanced Attack with 5 Ranges | Accuracy | 48.2% | 38% | 56.2% |
| | Average Changed Pixel Values | 3.9 | 3.8 | 3.9 |

example, irradiating part of the 2-dimensional RRAM area with infrared rays). This substantially improves the practicality and concealment of deploying the attack.

## ENHANCED FAULT-AWARE ADVERSARY ATTACK

The existing fault-aware adversary attack algorithm adds perturbations on a few pixels in adversary images. The amplitude of the perturbation is unlimited, which means a black pixel can be inversed as white in the grey image. As shown in **Figure 10**, such a unconstrainted modification leads to obvious manipulations in the attacked images. A typical phenomenon is the discontinued strokes of numbers, which can be easily detected and defended. The original intention of the full-ranged perturbation is to ensure that a sufficient change can be accumulated in the gradient ascent direction. Hence, we propose to increase the number of perturbations and limit the amplitude of each perturbation. This method ensures the trigger of misclassification by a large number of perturbations but each perturbation is less noticeable.

We denote the perturbation range as $[-\delta_{max}, \delta_{max}]$, then the full-ranged perturbation is within [-1.1] for MNIST images. As can be observed in **Table 2**, the proposed method achieves better attack effectiveness with a smaller average change in pixel values of an image. The new strategy not only reduces the number of discontinued strokes in the original position (e.g., '5' in the third row of **Figure 10**), but also motivates the algorithm to find a new misleading pattern (e.g., '1' in the third row of **Figure 10**), which explains the improvement in attack success rate. The images on the third row of **Figure 10** illustrate that the algorithm no longer creates obvious disconnection in handwritten figures. Compared to the images generated by the existing method, the ones generated by the proposed method achieve a similar appearance as the original images. Moreover, since the attackable images with different perturbation ranges do not always overlap, independently applying attacks with different ranged perturbations can further improve the attack effectiveness. For example, performing an attack of 10 perturbations within [-1,1] and 20 perturbations within [-0.35,0.35] results in a 17.2% improvement in the attack success rate without increasing the change of pixel values. Performing an attack by traverse through 10 perturbations within [-1,1] and 20 perturbations within [-0.1,0.1] [-0.2,0.2] [-0.3,0.3], and [-0.35,0.35] further reduce the accuracy to 40.6%. In addition, the enhanced attack is also deployed on targets with different fault types. As shown in **Table 3**, the method is applicable to both other hard faults (dual RRAM) and soft faults (normal variation and degraded precision).

## POTENTIAL DEFENSE METHODS

Since conventional fault-based adversary attacks aim to actively inject faults to selected parameters in memory by Trojan, some defense methods have been proposed to successfully prevent most of the threats (Liu et al., 2017c; Gu et al., 2017; Chen et al., 2018; Wang et al., 2019). However, they do not apply to the RRAM-targeted attack method for two reasons. First, some of those countermeasures can detect and remove the misleading by analyzing the results of testing classification. But the natural faults appear randomly in RRAM cells and hence are immune to such defense methods. Second, a fault-aware adversary attack assumes that the user simply transplants the neural network to the hardware and ignores the nonideality of the circuit while the defense methods require the full information of the prepared chip. Third, these defense methods are not designed to effectively defend the attack during the inference stage. Even if the full information of the nonideality is known and fixed, none of the defense methods can be deployed to prevent the faults/variations that occur during the operation or are injected *in-situ*.

Checking data integrity is another approach to solve/alleviate the fault-based adversary attack. However, popular data error detection/correction techniques, such as error-correction code (ECC), do not apply to the analog circuit. Besides, inherent faults are unfixable, and

knowing their existence cannot prevent the attack. As discussed in *Potential Defense Methods*, dynamically replacing unfixable RRAM cells with backup cells is not practical. Moreover, for the faults/variations that are fixable and occur during the operation, the user can repair/replace them by repetitively scanning the RRAM status. However, such a naive approach is inefficient and leads to unacceptable overheads in delay and power consumption. In addition, the designer can consider adding temperature (or other stimuli) sensors on the chip to monitor environmental temperature status and report to the user once an anomaly happens. However, this method is inapplicable to malicious stimuli that focus on partial RRAM, and it cannot solve the faults/variation caused by inherent nonideality. Finally, destroying the physical reading port of neuromorphic hardware can prevent the attacker from accessing fault/variation information, which is previously used to protect secure data in memory. Unfortunately, this scheme also blocks access to the device when an update needs to be performed on the neural network. As a result, it is only suitable for one-time training and stable digital memory with a long data retention time.

## CONCLUSION

In this paper, five faults/variation-based adversary attack models are developed based on detailed nonideal properties that exist in the RRAM crossbar array. By analyzing the device- and circuit-level phenomena, the scope of the attack is extended from simple and abstract binary hard faults to non-binary faults, randomly injected faults, uneven-distributed faults, and soft faults. The investigation reveals that adding faults to neural layers that are closer to input is much more efficient than those closer to output. More importantly, the quantification error and conductance variation are proven to be effective under the variation-aware attacks, in which the conductance variation achieves a 49.4% improvement in attack success rate compared to the attack based on original binary faults. Finally, an enhanced attack method is proposed to obtain more stealthy adversary images with up to 19.9% better attack success rates.

## DATA AVAILABILITY STATEMENT

Publicly available datasets were analyzed in this study. This data can be found here: http://yann.lecun.com/exdb/mnist/.

## AUTHOR CONTRIBUTIONS

LS designed and conducted most of the experiments. SJ provided the discussion of adversary attack background information and applications. FL revised this paper and contributed to data analysis. CP led this project.

## REFERENCES


An, H., An, Q., and Yi, Y. (2019). Realizing Behavior Level Associative Memory Learning through Three-Dimensional Memristor-Based Neuromorphic Circuits. *IEEE Trans. Emerg. Top. Comput. Intell.* 5 (4), 668–678. doi:10.1109/TETCI.2019.2921787

AWS AWS Deeplens(2022). Availabel at: https://aws.amazon.com/cn/deeplens/.

Beigi, M. V., and Memik, G. (2016). "TAPAS: Temperature-Aware Adaptive Placement for 3D Stacked Hybrid Caches," in Proceedings of the Second International Symposium on Memory Systems, Alexandria, VA, United States, October 3–6, 2016, 415–426.

Bhattacharjee, A., and Panda, P. (2020). Rethinking Non-idealities in Memristive Crossbars for Adversarial Robustness in Neural Networks. Available at: https://arxiv.org/pdf/2008.11298.pdf.

Breier, J., Hou, X., Jap, D., Ma, L., Bhasin, S., and Liu, Y. (2018). "Practical Fault Attack on Deep Neural Networks," in Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security, Toronto, Canada, October 15–19, 2018, 2204–2206. doi:10.1145/3243734.3278519

Burr, G. W., Shelby, R. M., Sidler, S., di Nolfo, C., Jang, J., Boybat, I., et al. (2015). Experimental Demonstration and Tolerancing of a Large-Scale Neural Network (165 000 Synapses) Using Phase-Change Memory as the Synaptic Weight Element. *IEEE Trans. Electron Devices* 62 (11), 3498–3507. doi:10.1109/ted.2015.2439635

Carlini, N., and Wagner, D. (2017). "Towards Evaluating the Robustness of Neural Networks," in 2017 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA, 22-26 May 2017 (IEEE), 39–57. doi:10.1109/sp.2017.49

Chen, B., Carvalho, W., Baracaldo, N., Ludwig, H., Edwards, B., Lee, T., et al. (2018). Detecting Backdoor Attacks on Deep Neural Networks by Activation Clustering. Available at: https://arxiv.org/abs/1811.03728.

Chen, C.-Y., Shih, H-C., Wu, C-W., Lin, C-H., Chiu, P-F., Sheu, S-S., et al. (2014). RRAM Defect Modeling and Failure Analysis Based on March Test and a Novel Squeeze-Search Scheme. *IEEE Trans. Comput.* 64 (1), 180–190. doi:10.1109/TC.2014.12

Chen, J., and Ran, X. (2019). Deep Learning with Edge Computing: A Review. *Proc. IEEE* 107 (8), 1655–1674. doi:10.1109/jproc.2019.2921977

Chen, P.-Y., Peng, X., and Yu, S. (2017). "NeuroSim+: An Integrated Device-To-Algorithm Framework for Benchmarking Synaptic Devices and Array Architectures," in 2017 IEEE International Electron Devices Meeting (IEDM), San Francisco, CA, USA, 2-6 Dec. 2017 (IEEE), 6.1. 1–6.1. 4.

Chinchali, S., Hu, P., Chu, T., Sharma, M., Bansal, M., Misra, R., et al. (2018). "Cellular Network Traffic Scheduling with Deep Reinforcement Learning," in Thirty-second AAAI conference on artificial intelligence, New Orleans, LA, United States, February 2–7, 2018.

Fadeev, A. V., and Rudenko, K. V. (2021). To the Issue of the Memristor's HRS and LRS States Degradation and Data Retention Time. *Russ. Microelectron.* 50 (5), 311–325. doi:10.1134/s1063739721050024

Giacomin, E., Greenberg-Toledo, T., Kvatinsky, S., and Gaillardon, P.-E. (2018). A Robust Digital RRAM-Based Convolutional Block for Low-Power Image Processing and Learning Applications. *IEEE Trans. Circuits Syst. I Regul. Pap.* 66 (2), 643–654. doi:10.1109/TC.2014.12

Goodfellow, I. J., Shlens, J., and Szegedy, C. (2014). Explaining and Harnessing Adversarial Examples. Available at: https://arxiv.org/abs/1412.6572.

Gu, T., Dolan-Gavitt, B., and Garg, S. (2017). Badnets: Identifying Vulnerabilities in the Machine Learning Model Supply Chain. Available at: https://arxiv.org/abs/1708.06733.

Ha, K., Chen, Z., Hu, W., Richter, W., Pillai, P., and Satyanarayanan, M. (2014). "Towards Wearable Cognitive Assistance," in Proceedings of the 12th annual international conference on Mobile systems, applications, and services, Bretton Woods, NH, United States, June 16–19, 2014, 68–81. doi:10.1145/2594368.2594383

He, Z., Lin, J., Ewetz, R., Yuan, J.-S., and Fan, D. (2019). "Noise Injection Adaption: End-To-End ReRAM Crossbar Non-ideal Effect Adaption for Neural Network Mapping," in Proceedings of the 56th Annual Design Automation Conference 2019, Las Vegas, NV, USA, 2-6 June 2019, 1–6.

Hsu, T.-H., Chiu, Y-C., Wei, W-C., Lo, Y-C., Lo, C-C., Liu, R-S., et al. (2019). "AI Edge Devices Using Computing-In-Memory and Processing-In-Sensor: from

System to Device," in 2019 IEEE International Electron Devices Meeting (IEDM), San Francisco, CA, USA, 7-11 Dec. 2019 (IEEE), 22.5. 1–22.5. 4. doi:10.1109/iedm19573.2019.8993452

Hu, M., Graves, C. E., Li, C., Li, Y., Ge, N., Montgomery, E., et al. (2018). Memristor-Based Analog Computation and Neural Network Classification with a Dot Product Engine. *Adv. Mat.* 30 (9), 1705914. doi:10.1002/adma.201705914

Hung, C.-C., Ananthanarayanan, G., Bodik, P., Golubchik, L., Yu, M., Bahl, P., et al. (2018). "Videoedge: Processing Camera Streams Using Hierarchical Clusters," in 2018 IEEE/ACM Symposium on Edge Computing (SEC), Seattle, WA, USA, 25-27 Oct. 2018 (IEEE), 115–131.

Jeong, Y., Kim, S., and Lu, W. D. (2015). Utilizing Multiple State Variables to Improve the Dynamic Range of Analog Switching in a Memristor. *Appl. Phys. Lett.* 107 (17), 173105. doi:10.1063/1.4934818

Jiang, H., Han, L., Lin, P., Wang, Z., Jang, M. H., Wu, Q., et al. (2016). Sub-10 Nm Ta Channel Responsible for Superior Performance of a HfO2 Memristor. *Sci. Rep.* 6 (1), 28525–28528. doi:10.1038/srep28525

Jo, S. H., Chang, T., Ebong, I., Bhadviya, B. B., Mazumder, P., and Lu, W. (2010). Nanoscale Memristor Device as Synapse in Neuromorphic Systems. *Nano Lett.* 10 (4), 1297–1301. doi:10.1021/nl904092h

Kaneko, Y., Nishitani, Y., and Ueda, M. (2014). Ferroelectric Artificial Synapses for Recognition of a Multishaded Image. *IEEE Trans. Electron Devices* 61 (8), 2827–2833. doi:10.1109/ted.2014.2331707

Kim, S., Ishii, M., Lewis, S., Perri, T., BrightSky, M., Kim, W., et al. (2015). "NVM Neuromorphic Core with 64k-Cell (256-by-256) Phase Change Memory Synaptic Array with On-Chip Neuron Circuits for Continuous *In-Situ* Learning," in 2015 IEEE international electron devices meeting (IEDM), Washington, DC, USA, 7-9 Dec. 2015 (IEEE), 17.1. 1–17.1. 4. doi:10.1109/iedm.2015.7409716

Kim, Y., Zhang, Y., and Li, P. (2012). "A Digital Neuromorphic VLSI Architecture with Memristor Crossbar Synaptic Array for Machine Learning," in 2012 IEEE International SOC Conference, Niagara Falls, NY, USA, 12-14 Sept. 2012 (IEEE), 328–333.

Lin, P., Li, C., Wang, Z., Li, Y., Jiang, H., Song, W., et al. (2020). Three-dimensional Memristor Circuits as Complex Neural Networks. *Nat. Electron* 3 (4), 225–232. doi:10.1038/s41928-020-0397-9

Liu, C., Hu, M., Strachan, J. P., and Li, H. (2017). "Rescuing Memristor-Based Neuromorphic Design with High Defects," in 2017 54th ACM/EDAC/IEEE Design Automation Conference (DAC), Austin, TX, USA, 18-22 June 2017 (IEEE), 1–6. doi:10.1145/3061639.3062310

Liu, L., Li, H., and Gruteser, M. (2019). "Edge Assisted Real-Time Object Detection for Mobile Augmented Reality," in The 25th Annual International Conference on Mobile Computing and Networking, Los Cabos, Mexico, October 21–25, 2019, 1–16. doi:10.1145/3300061.3300116

Liu, P., You, Z., Kuang, J., Hu, Z., Duan, H., and Wang, W. (2016). Efficient March Test Algorithm for 1T1R Cross-bar with Complete Fault Coverage. *Electron. Lett.* 52 (18), 1520–1522. doi:10.1049/el.2016.1693

Liu, S., Wang, Y., Fardad, M., and Varshney, P. K. (2018). A Memristor-Based Optimization Framework for Artificial Intelligence Applications. *IEEE Circuits Syst. Mag.* 18 (1), 29–44. doi:10.1109/mcas.2017.2785421

Liu, Y., Ma, S., Aafer, Y., Lee, W-C., and Zhai, J. (2017). "Trojaning Attack on Neural Networks," in Conference: Network and Distributed System Security Symposium, San Diego, CA, United States, February 26–March 1, 2017. doi:10.14722/ndss.2018.23300

Liu, Y., Wei, L., Luo, B., and Xu, Q. (2017). "Fault Injection Attack on Deep Neural Network," in 2017 IEEE/ACM International Conference on Computer-Aided Design (ICCAD), Irvine, CA, USA, 13-16 Nov. 2017 (IEEE), 131–138. doi:10.1109/iccad.2017.8203770

Lv, H., Xu, X., Liu, H., Liu, R., Liu, Q., Banerjee, W., et al. (2015). Evolution of Conductive Filament and its Impact on Reliability Issues in Oxide-Electrolyte Based Resistive Random Access Memory. *Sci. Rep.* 5 (1), 7764–7766. doi:10.1038/srep07764

Lv, H., Li, B., Wang, Y., Liu, C., and Zhang, L. (2021). "VADER: Leveraging the Natural Variation of Hardware to Enhance Adversarial Attack," in 2021 26th Asia and South Pacific Design Automation Conference (ASP-DAC), Tokyo, Japan, 18-21 Jan. 2021 (IEEE), 487–492.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2017). Towards Deep Learning Models Resistant to Adversarial Attacks. Available at: https://arxiv.org/abs/1706.06083.

M. A. E. Challenge(2017). Available at: https://github.com/MadryLab/mnist_challenge#mnist-adversarial-examples-challenge.

Mohammadi, M., Al-Fuqaha, A., Sorour, S., and Guizani, M. (2018). Deep Learning for IoT Big Data and Streaming Analytics: A Survey. *IEEE Commun. Surv. Tutorials* 20 (4), 2923–2960. doi:10.1109/comst.2018.2844341

Münch, C., Bishnoi, R., and Tahoori, M. B. (2019). "Reliable In-Memory Neuromorphic Computing Using Spintronics," in Proceedings of the 24th Asia and South Pacific Design Automation Conference, Tokyo Odaiba Waterfront, January 21–24, 2019, 230–236.

Papernot, N., McDaniel, P., Goodfellow, I., Jha, S., Celik, Z. B., and Swami, A. (2017). "Practical Black-Box Attacks against Machine Learning," in Proceedings of the 2017 ACM on Asia conference on computer and communications security, San Jose, CA, United States, May 22–26, 2017, 506–519. doi:10.1145/3052973.3053009

Papernot, N., McDaniel, P., Wu, X., Jha, S., and Swami, A. (2016). "Distillation as a Defense to Adversarial Perturbations against Deep Neural Networks," in 2016 IEEE symposium on security and privacy (SP), San Jose, CA, USA, 22-26 May 2016 (IEEE), 582–597. doi:10.1109/sp.2016.41

Prezioso, M., Kataeva, I., Merrikh-Bayat, F., Hoskins, B., Adam, G., Sota, T., et al. (2015). "Modeling and Implementation of Firing-Rate Neuromorphic-Network Classifiers with Bilayer Pt/Al2O3/TiO2− x/Pt Memristors," in 2015 IEEE International Electron Devices Meeting (IEDM), Washington, DC, USA, 7-9 Dec. 2015 (IEEE), 17.4. 1–17.4. 4. doi:10.1109/iedm.2015.7409719

Rakin, A. S., He, Z., and Fan, D. (2020). "Tbt: Targeted Neural Network Attack with Bit Trojan," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13-19 June 2020, 13198–13207. doi:10.1109/cvpr42600.2020.01321

Rakin, A. S., He, Z., Li, J., Yao, F., Chakrabarti, C., and Fan, D. (2021). T-bfa: Targeted Bit-Flip Adversarial Weight Attack. *IEEE Trans. Pattern Analysis Mach. Intell.* doi:10.1109/TPAMI.2021.3112932

Singh, A., Diware, S., Gebregiorgis, A., Bishnoi, R., Catthoor, F., Joshi, R. V., et al. (2021). "Low-power Memristor-Based Computing for Edge-AI Applications," in 2021 IEEE International Symposium on Circuits and Systems (ISCAS), Daegu, Korea, 22-28 May 2021 (IEEE), 1–5. doi:10.1109/iscas51556.2021.9401226

Temam, O. (2012). "A Defect-Tolerant Accelerator for Emerging High-Performance Applications," in 2012 39th Annual International Symposium on Computer Architecture (ISCA)9-13 June 2012, Portland, OR, USA (IEEE), 356–367. doi:10.1145/2366231.2337200

Tunali, O., and Altun, M. (2016). Permanent and Transient Fault Tolerance for Reconfigurable Nano-Crossbar Arrays. *IEEE Trans. Computer-Aided Des. Integr. Circuits Syst.* 36 (5), 747–760.

Wang, B., Yao, Y., Shan, S., Li, H., Viswanath, B., Zheng, H., et al. (2019). "Neural Cleanse: Identifying and Mitigating Backdoor Attacks in Neural Networks," in 2019 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 19-23 May 2019 (IEEE), 707–723. doi:10.1109/sp.2019.00031

Wu, M.-C., Chen, J.-Y., Ting, Y.-H., Huang, C.-Y., and Wu, W.-W. (2021). A Novel High-Performance and Energy-Efficient RRAM Device with Multi-Functional Conducting Nanofilaments. *Nano Energy* 82, 105717. doi:10.1016/j.nanoen.2020.105717

Xia, L., Huangfu, W., Tang, T., Yin, X., Chakrabarty, K., Xie, Y., et al. (2017). Stuck-at Fault Tolerance in RRAM Computing Systems. *IEEE J. Emerg. Sel. Top. Circuits Syst.* 8 (1), 102–115.

Xia, L., Liu, M., Ning, X., Chakrabarty, K., and Wang, Y. (2018). Fault-tolerant Training Enabled by On-Line Fault Detection for RRAM-Based Neural Computing Systems. *IEEE Trans. Computer-Aided Des. Integr. Circuits Syst.* 38 (9), 1611–1624.

Yao, P., Wu, H., Gao, B., Tang, J., Zhang, Q., Zhang, W., et al. (2020). Fully Hardware-Implemented Memristor Convolutional Neural Network. *Nature* 577 (7792), 641–646. doi:10.1038/s41586-020-1942-4

Yeo, I., Chu, M., Gi, S.-G., Hwang, H., and Lee, B.-G. (2019). Stuck-at-fault Tolerant Schemes for Memristor Crossbar Array-Based Neural Networks. *IEEE Trans. Electron Devices* 66 (7), 2937–2945. doi:10.1109/ted.2019.2914460

Yu, S., Chen, P.-Y., Cao, Y., Xia, L., Wang, Y., and Wu, H. (2015). "Scaling-up Resistive Synaptic Arrays for Neuro-Inspired Architecture: Challenges and Prospect," in 2015 IEEE International Electron Devices Meeting (IEDM), Washington, DC, USA, 7-9 Dec. 2015 (IEEE), 17.3. 1–17.3. 4. doi:10.1109/iedm.2015.7409718

Zhang, T., Chowdhery, A., Bahl, P., Jamieson, K., and Banerjee, S. (2015). "The Design and Implementation of a Wireless Video Surveillance System," in Proceedings of the 21st Annual International Conference on Mobile Computing and Networking, New York, NY, United States, September 7–11, 2015, 426–438. doi:10.1145/2789168.2790123

Zhou, Z., Chen, X., Li, E., Zeng, L., Luo, K., and Zhang, J. (2019). Edge Intelligence: Paving the Last Mile of Artificial Intelligence with Edge Computing. *Proc. IEEE* 107 (8), 1738–1762. doi:10.1109/jproc.2019.2918951

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

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