FairXbar: Improving the Fairness of Deep Neural Networks with Non-Ideal In-Memory Computing Hardware

Sohan Salahuddin Mugdho*, Yuanbo Guo†, Ethan G. Rogers*, Weiwei Zhao*, Yiyu Shi† and Cheng Wang*
Department of Electrical and Computer Engineering
Iowa State University of Science and Technology, Ames, IA 50010
Computer Science and Engineering
University of Notre Dame, Notre Dame, IN 46556
*Corresponding Author: chengw@iastate.edu

***Abstract*—While artificial intelligence (AI) based on deep neural networks (DNN) has achieved near-human performance in various cognitive tasks, such data-driven models are known to exhibit implicit bias against specific subgroups, leading to fairness issues. Most existing methods for improving model fairness only consider software-based optimizations, while the impact of hardware is largely unexplored. In this work, we investigate the impact of underlying hardware technology on AI fairness as we deploy DNN-based medical diagnosis algorithms onto in-memory computing hardware accelerators. Based on our newly developed framework that characterizes the importance of DNN weight parameters to fairness, we demonstrate that device variability-induced non-idealities such as stuck-at faults and noises due to variation can be exploited to deliver improved fairness (up to 32% improvement) with significantly reduced trade-off (less than 1% loss) of the overall accuracy. We additionally develop a hardware non-idealities-aware training methodology that further mitigates the bias between unprivileged and privileged demographic groups in our experiments on skin lesion diagnosis datasets. Our work suggests exciting opportunities for leveraging the hardware attributes in a cross-layer co-design to enable equitable and fair AI.**

***Index Terms*—AI Fairness, Crossbar Nonidealities, Deep Neural Networks, Hardware Accelerators, In-Memory Computing**

I. Introduction

Deep neural networks (DNNs) have achieved remarkable advances, showing near-human cognitive capabilities across diverse domains like computer vision and medical diagnosis to language processing [1] [2]–[4], and driving the integration of DNN-based artificial intelligence (AI) into the decision-making of critical tasks, such as identifying tumor types, qualification assessment of job candidates, and deciding detention or release of people under investigation/probation [5] [6] [7]. Despite their ability to learn complex patterns, the "black-box"-like DNN models remain prone to biases against certain subgroups with specific demographic features like skin tone or gender. Significant societal issues may arise if such biased algorithms are deployed in society. Recent investigations showed that even after dataset imbalance is addressed, the fairness issue of DNN algorithms remains, indicating the fairness issue is challenging and requires novel solutions [8].

The rapid development of DNNs also leads to a surge of hardware accelerators with diverse architectures to accommodate the processing of data-intensive computations [9]–[11]. Most AI accelerators aim to improve the computational efficiency of matrix-vector multiplications (MVM) that dominate the operations in most DNN workloads [12]. Particularly, emerging architectures such as in-memory-computing (IMC) accelerators tackle the notorious von Neumann memory bottleneck by enabling in-memory and parallel MVM based on crossbar arrays [13]. Since IMC accelerators are erroneous due to analog-domain accumulation, extensive hardware-software co-design has been conducted to achieve high efficiency while maintaining satisfactory functional accuracy. However, most optimization of IMC hardware focused on evaluating the average accuracy, while possible accuracy disparity among different subgroups is largely neglected.

In this work, we explore the impacts of hardware variability from emerging IMC hardware on the fairness of DNN models. While seeking novel ways of tackling algorithmic biases of DNNs, we go beyond the conventional algorithmic optimizations and venture into utilizing hardware errors to make the deployed neural network robust against unfairness. As our framework takes advantage of hardware non-idealities to improve fairness, our proposed methods have the potential to be applied not only on IMC hardware but also on other AI hardware with tunable variability. The major contributions of this paper are summarized as follows:

- We show that hardware non-idealities from IMC accelerator variabilities can be exploited to mitigate DNN algorithmic bias. Our hardware-aware error injection analysis reveals that addressing unfairness requires decoupling the impact of errors on *average* accuracy and *bias against unprivileged subgroups*.
- We develop a salience-based framework to rank DNN weights by their respective importance to accuracy and bias. The proposed ranking-based inhomogeneous error injection scheme introduces elevated errors into target model weights, achieving a trade-off that noticeably re-

978-3-9826741-0-0/DATE25/© 2025 EDAA

duces bias while minimizing accuracy loss.

- We integrate saliency-based ranking analysis with a fairness-aware fine-tuning (retraining) scheme to further enhance the impact of hardware attributes. Retraining with error injection informed by the combination of accuracy and fairness saliency ranking significantly reduces bias against unprivileged subgroups by $\geq 30\%$ while maintaining average prediction accuracy.

II. Background

A. The Fairness Challenge of DNN

Concerns about DNN fairness have grown alongside advances in deep learning. Studies show persistent gender and skin-type biases in commercial AI systems, with facial-analysis software error rates of 0.8% for light-skinned men and 34.7% for dark-skinned women [14]. Similar racial disparities appear in AI-based skin condition diagnoses [15]. These fairness issues are magnified in biomedical applications, impacting societal well-being [16]–[18].

It is vital to look for commonly used fairness metrics to better understand and discuss fairness. Equalized odds (EOdds) and Equalized opportunity (EOpp0) [19] are two such metrics. EOdds is the sum of the true positive rate difference and the false positive rate difference measured on groups of sensitive attributes, while the EOpp0 is defined as the difference of true negative rate values. By noting True Positive, False Negative, True Negative, and False Positive values of class k and group c as TP_k^c, FN_k^c, TN_k^c, and FP_k^c, we obtain for class k and group c, $TPR_k^c = \frac{TP_k^c}{TP_k^c+FN_k^c}$, $TNR_k^c = \frac{TN_k^c}{TN_k^c+FP_k^c}$, $FPR_k^c = \frac{FP_k^c}{TN_k^c+FP_k^c}$. Thus, we find the EOdds and EOpp0 metrics as:

$$EOdds = \sum_{k=1}^{K} \left| TPR_k^1 - TPR_k^0 + FPR_k^1 - FPR_k^0 \right|, \quad (1)$$

$$EOpp0 = \sum_{k=1}^{K} \left| TNR_k^1 - TNR_k^0 \right|, \quad (2)$$

where, $c = 0$ for the unprivileged group, and $c = 1$ for the privileged group. For both metrics, smaller values indicate more fairness. On the other hand, the overall performance of AI models is represented by the widely used F1-score, which is robust against class imbalance. Additionally, the F1-score is measured as the harmonic mean of Precision and Recall that balances the trade-offs between False Negatives and False Positives. We find the F1-score as,

$$F1 - score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}}, \quad (3)$$

where, $Precision = \frac{TP}{TP+FP}$ and $Recall = \frac{TP}{TP+FN}$, with TP, FP, and FN being the True Positive, False Positive, and False Negative values across all classes and groups. A higher F1-score indicates better overall performance.

Fig. 1 illustrates the fairness challenge exemplified in unbalanced prediction accuracy (F1-score) depending on skin tones. "No fairness awareness" leads to good *average* F1 accuracy but large differences between the two groups (light/dark skin tones), while prioritizing fairness is prone to have improved fairness at the cost of low overall accuracy ("Poor Fairness awareness"). We aim to develop tools to enable fair and accurate models ("Good Fairness Awareness"), as shown in the third column.

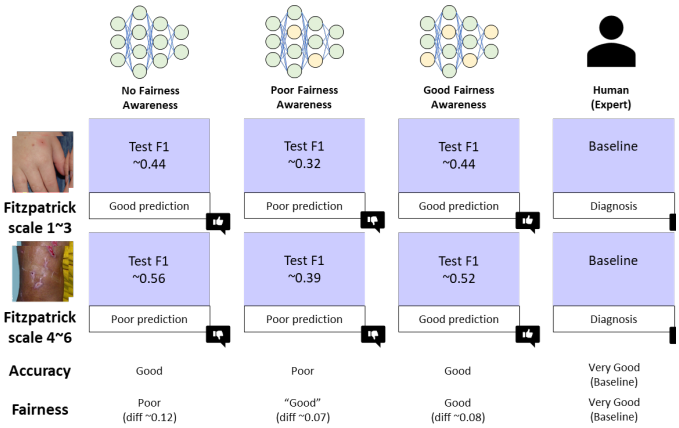

Fig. 1: Illustrating the fairness problem of DNN. Samples images from the Fitzpatrick-17k [20] dataset are used as examples. The DNN model used to measure F1 results is modified accordingly to demonstrate different fairness awareness.

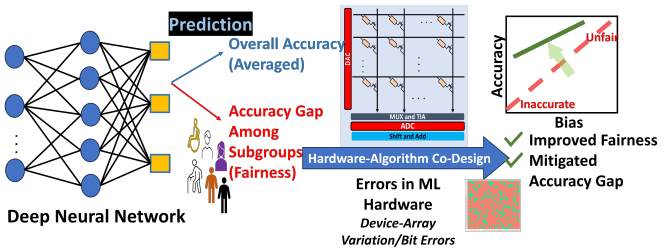

Fig. 2: Overview of FairXbar. Hardware errors of non-ideal in-memory computing crossbars are exploited to address the DNN fairness problem.

B. In-Memory Computing for DNN Acceleration

Crossbar-based IMC has emerged as a promising hardware platform for accelerating DNN processing [10], [13]. In an IMC computing core, as shown in Fig. 2, the data-intensive matrix-vector multiplications (MVM) can be computed with massive parallelism in the analog domain based on the accumulated currents along the vertical lines following Kirchoff's current law (KCL). IMC architectures based on emerging non-volatile memories(NVMs) alleviate the von Neumann memory bottleneck associated with the movement of weights, providing a promising solution for low-power DNN inference acceleration. To accommodate large-scale DNN workloads, spatial tiling of multiple computing cores based on crossbar arrays is exploited to map large matrices and convolutional kernels [21]. Furthermore, bit slicing of input and weights can be implemented to map high-resolution workloads onto low-precision hardware primitives [10].

C. Variability-induced Non-ideality in Emerging Hardware

While IMC demonstrates significant potential, analog MVM suffers from numerous device/circuit variabilities. During

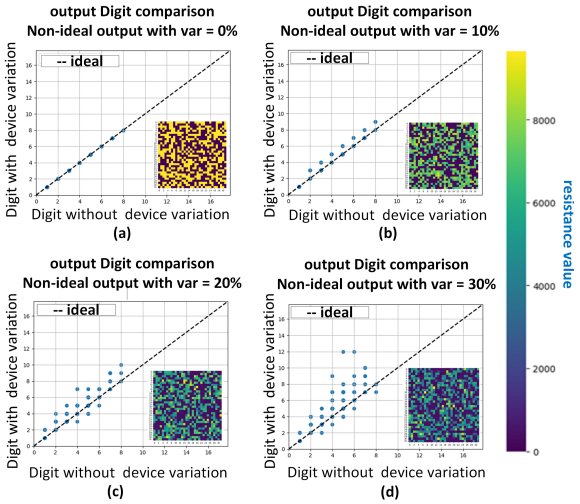

Fig. 3: Comparison of digits under varying levels of device variation modeled using Gaussian distribution based on σ'_G.(a) σ'_G=0%; (b) σ'_G=10%; (c) σ'_G=20%; (d) σ'_G=30%.

DNN inference, the predominant sources of error are the stuck-at-fault errors and weight errors due to device cycle-to-cycle variations [22].

Stuck-at fault. Stuck-at faults (SAF) occur when an NVM cell (such as a resistive random access memory ReRAM) becomes permanently locked in a specific resistance state, rendering it unable to alter its resistance in Write Operations [23]. Stuck-at-1 (Stuck-at-0) describes the occurrence of a memory device permanently set to a high (low) resistance state. In this study, SAFs are emulated as stochastic (but permanent) bit-flips following a specific pattern [24], with equal flipping rates for 0s and 1s.

Noises due to variation. The noise in DNN weights due to variation in the weight storage devices, referred to as "*Noise due to variation*" throughout the paper, is modeled as [25]:

$$w = w_0 + \Delta w, \Delta w \sim \mathcal{N}(0, \sigma). \tag{4}$$

Here w_0 represents the ideal weight value, and w represents the weight value influenced by device variation expressed as a zero-mean Gaussian noise Δw. For a general multi-bit NVM device, a weight can be stored with bit slicing across M NVM devices where each device holds N-bits of the weight. The standard deviation σ of the Gaussian noise Δw under such bit slicing configuration can be modeled as [26],

$$\sigma = \sigma'_G \sqrt{1 + 2^{2(\frac{N}{M}-1)}}. \tag{5}$$

Here, σ'_G, mentioned as the Noise rate throughout the rest of the paper, is the device-level standard deviation. Such device variations lead to errors in the MVM output. Fig. 3 illustrates the comparison of ideal and non-ideal MVM output under varying levels of device variation. Note that the plotted data points are MVM output after the crossbar currents are binned to digital values based on ideal analog-digital conversion. Fig. 3 (a)-(d) clearly demonstrates that the output errors (the vertical spreading of output values) grow with increased device variation from 0% to 30%.

Tunable mitigation of variability-induced non-ideality. It is important to note that numerous circuit/system-level techniques have been developed to mitigate the non-idealities due to device variability. SAF errors can be mitigated via error correction, post-processing and remapping methods [27]–[29], while device variation can be reduced by write-verify and other hardware-software co-design methods [30]–[33]. In hardware with error-prone NVM cells, such methods enable selectively mitigating the error of the target NVM cells, effectively applying the error to the target weights. Our proposed framework will build upon these techniques to adjust the type and severity of error injection to improve DNN fairness. Besides SAF and noises due to device variation, other non-idealities, such as weight drift and variations in voltage/current offset in ADC, and errors beyond the hardware could also be investigated to explore their impacts on the DNN fairness using our framework.

III. Related Work

The Fairness of neural networks has become an emerging topic that received growing attention recently [8] [34]. Most of the efforts on addressing the DNN fairness focused on algorithmic consideration [19] [35] and addressing the dataset-related limitation [36]. Specifically, methods based on shortcut learning, subsampling, gradient reversal, and adversarial learning have been studied aiming to mitigate unfairness [37]–[39]. Further algorithmic approaches based on bias-aware training, generative models, and differential privacy have been explored for improving fairness [40]–[42]. Additionally, the impacts of DNN model size, quantization, and pruning on fairness have been investigated to develop more efficient and equitable DNN models. [43]–[45]. In terms of software implementation of DNN, It has been shown that the stochasticity generated from the different Python compilers and deep learning libraries may lead to variance of model accuracy and sizeable gaps among different classification groups [46]. Based on the observation that a low-level machine-dependent mechanism of generating stochasticity plays a key role in introducing the variation of DNN outcome, recently, the training accuracy and bias were assessed across multiple GPU platforms to gain a better understanding of the hardware sensitivity on bias [47]. However, how the hardware components directly interact with DNN fairness remains largely unexplored. Compared to the previous works, we, for the first time, venture into the device-circuit level to investigate the direct interaction of hardware and DNN fairness. In this work, we first analyze the impact of variability in emerging IMC hardware on fairness and subsequently develop our methodology to exploit hardware variability to *mitigate* the algorithmic-level fairness issue.

IV. Evaluation Methodology

We aim to reduce performance disparity (bias) between the unprivileged and privileged groups by letting device non-idealities affect weights with high fairness impact but low accuracy impact, based on the F1-score. In practice, all devices

in the IMC hardware would possess the desired non-idealities. After identifying the target weights for fairness in software, mitigation methods for variabilities, as discussed in Sec. II-C, need to be applied to the devices storing non-target weights, ensuring only the selected weights are injected with non-idealities. We evaluate the proposed framework in software by simulating the hardware non-idealities based on the discussion in Sec. II-C.

The saliency [45] analysis is utilized to indicate the importance of each weight for accuracy and/or fairness. Based on this, we target weights for error injection. First, we define a dataset $D = \{x_i, y_i, c_i\}$, $i \in \{1, ..., N\}$, where x_i is the input image, y_i is the class label, c_i is the sensitive attribute (skin tone, gender, etc.). Starting from a pre-trained model $F(\theta, x)$ that makes a prediction $y_i = F(\theta, x_i)$ with weights θ, we adjust only a subset of the weights to reduce the gap of the accuracy of predictions $F(\theta, x)$ when inputs are from groups with different sensitive attribute c. In this paper, we focus on one sensitive attribute based on the dataset, skin tone or gender, for which $c_i \in \{0, 1\}$, where $c = 0$ ($c = 1$) represents the unprivileged group (privileged group).

A. Saliency Analysis

Saliency measures the sensitivity of prediction accuracy change in response to some change in a weight. Given a pre-trained model $F(\theta, x)$, and its objective function $\mathcal{L}$, the change in objective function $\mathcal{L}$ via some variation in weights Θ can be approximated by the Taylor series expansion [48]:

$$\Delta\mathcal{L} = \mathcal{L}(D|\Theta = \Theta + \Delta\Theta) - \mathcal{L}(D)$$
$$= -\sum_i g_i\theta_i + \frac{1}{2}\sum_i h_{ii}\theta_i^2 + \frac{1}{2}\sum_{i\neq j} h_{ij}\theta_i\theta_j + O(||\Delta\Theta||^3).$$

Given a converged pre-trained model, the gradient $g_i = \frac{\partial\mathcal{L}}{\theta_i}$ will be close to 0. $h_{ii} = \frac{\partial^2\mathcal{L}}{\partial^2\theta_i}$ is the second derivative of weight θ_i, which can be found from the diagonal components of Hessian matrix H. Assuming that errors are injected independently into the weights, we will neglect the third term contributed from the off-diagonal components of Hessian. Finally, we have

$$\Delta\mathcal{L} = \frac{1}{2}\sum_i h_{ii}\theta_i^2, \tag{6}$$

where $\frac{1}{2}h_{ii}\theta_i^2$ is the saliency of θ_i, which represents the impact of error after applying variation to θ_i.

B. Weight Selection using Saliency

Some weights can possess more importance (saliency) than others [45]. Making those weights prone to device non-idealities (i.e., stuck-at-fault error, Noise due to variation, etc.) can help reduce the accuracy disparity between the unprivileged and the privileged groups and achieve fairness. The following fairness-aware saliency computation helps rank the weights important to the privileged group but unimportant to the unprivileged group [45]:

$$\min_\Theta \mathcal{J} = \Delta\mathcal{L}_{c=0}(\Theta) - \beta\Delta\mathcal{L}_{c=1}(\Theta) = \sum_i s_i, \tag{7}$$

where,

$$s_i = \frac{1}{2}h_{ii}^0\theta_i^2 - \beta\cdot\frac{1}{2}h_{ii}^0\theta_i^2. \tag{8}$$

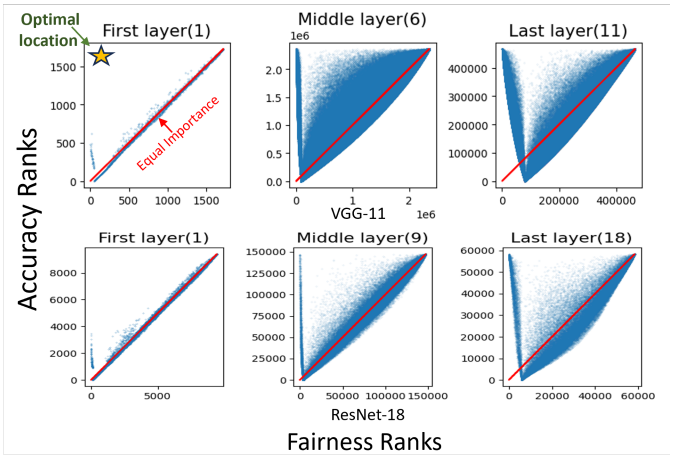

Fig. 4: Distribution of weights ranked based on accuracy vs fairness for the first, middle, and last layers of VGG-11 and ResNet-18.

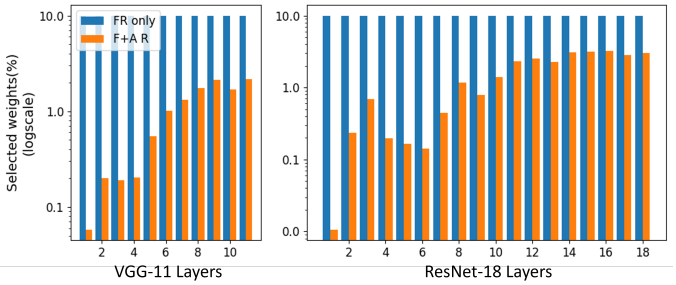

Fig. 5: Fraction of targeted weights ranked by only considering fairness (FR only) and ranked by considering both accuracy and fairness (F+A R).

By minimizing $\mathcal{J}$, we essentially minimize $\Delta\mathcal{L}_{c=0}(\Theta)$ and maximize $\Delta\mathcal{L}_{c=1}(\Theta)$, which results in identifying weights significant to the privileged group but negligible to the unprivileged group, i.e., important to fairness. Here, β is a hyper-parameter that controls the trade-off between $\Delta\mathcal{L}_{c=0}(\Theta)$ and $\Delta\mathcal{L}_{c=1}(\Theta)$.

Although solving Eq. 7 leads to identifying weights important to fairness, there is always a trade-off between fairness and accuracy. Fig. 4 shows the distribution of weights ranked on Eq. 9 vs weights ranked on Eq. 7. The weights are from the layers of the VGG11 [49] and ResNet-18 [2] pre-trained on the Fitzpatrick-17k dataset [20], [50], where smaller values of rank represent higher importance. Blindly applying variation to all of these weights leads to a suboptimal trade-off between accuracy and fairness.

C. Improving the trade-off

To improve the aforementioned accuracy-fairness trade-off, we define a second objective:

$$\min_\Theta \mathcal{K} = \Delta\mathcal{L}_{c=\{0,1\}}(\Theta) = \sum_i a_i, \tag{9}$$

where,

$$a_i = \frac{1}{2}h_{ii}^0\theta_i^2 + \frac{1}{2}h_{ii}^0\theta_i^2. \tag{10}$$

Minimizing Eq. 9 essentially ranks the weights based on their importance in the overall accuracy of the model. As shown in

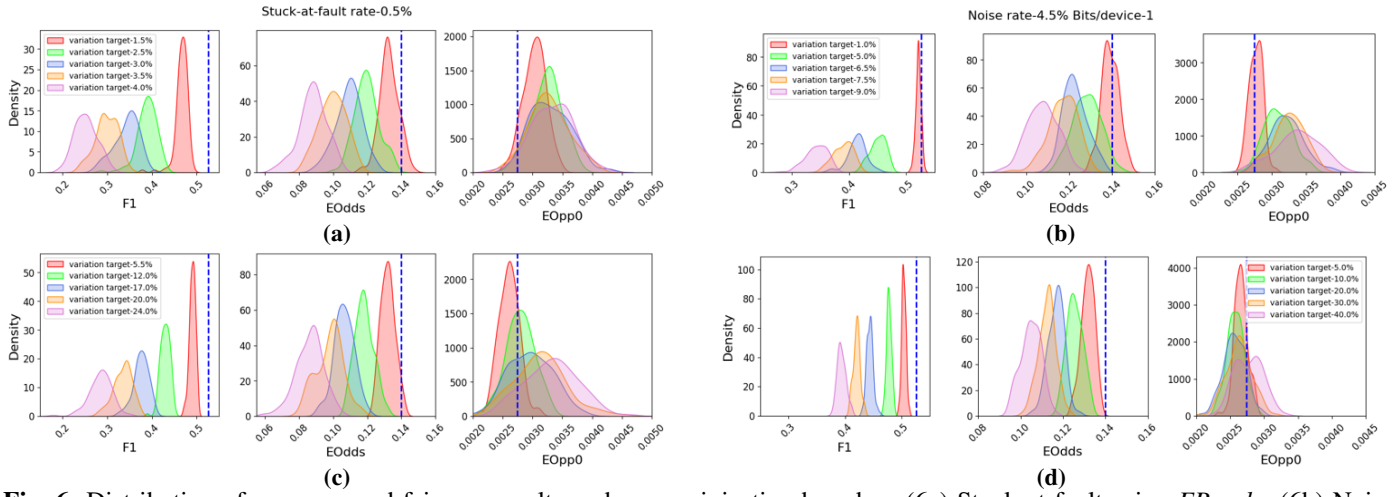

Fig. 6: Distribution of accuracy and fairness results under error injection based on (6a) Stuck-at-fault using *FR only*; (6b) Noise due to variation using *FR only*; (6c) Stuck-at-fault using *F+A R*; and (6d) Noise due to variation using *F+A R*.

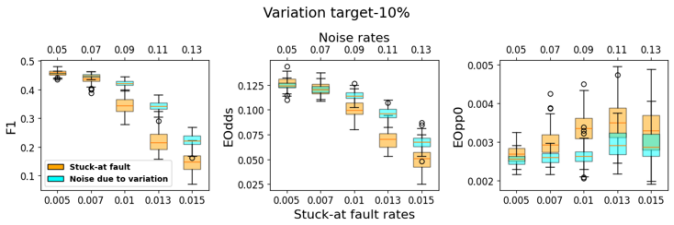

Fig. 7: Sensitivity of model performance (accuracy, fairness) to the intensity of applied stuck-at fault error and noise due to variation. For both cases, the variation target is 10%.

Fig. 4, solving Eq. 7 alone targets a large portion of weights that affect both fairness and accuracy. Based on the insight from Fig. 4, we rank the weights considering both fairness and accuracy following Eq. 9 and 7, aiming to target the weights that are significant to fairness but negligible to accuracy (*F+A R* configuration). As shown in Fig. 5, this method results in an inhomogeneous application of device variations to the weights across the layers, contrary to the configuration where we identify weights based only on fairness, (*FR only*), which results in targeting a fixed portion of weights in all layers.

D. Retraining Scheme

To improve the trade-off further, we explore two retraining schemes: 1. Variability-aware retraining (VA-Retrain), where we only consider the non-ideal behaviors, and 2. Variability+Fairness-aware retraining (VFA-Retrain), where besides the non-ideal behavior, we also consider fairness in the loss function. The retraining procedure iterates through the following general steps:

1) Sample mini-batches from the unprivileged and the privileged groups separately.
2) Compute the second derivatives h_{ii} of each weight with respect to the unprivileged and privileged groups.
3) Compute the fairness and accuracy saliencies s_i and a_i, and rank the weights based on the two categories, i.e., fairness and accuracy.
4) After each back-propagation, apply bit flips/noises in software to emulate the device errors (i.e., stuck-at fault or noise due to variation) to weights within the top $x\%$ on the fairness ranks (FR) and bottom $x\%$ on the accuracy ranks (AR).
5) Repeat steps 1-4 until some target (accuracy or fairness) is achieved.

For the Variability+Fairness-aware retraining scheme, we explore a fairness-aware objective function during retraining.

$$\min_{\theta} \mathcal{L}(\theta) = (1-\alpha)\cdot\mathcal{L}(\theta)_{c=\{0,1\}} + \alpha\cdot\left|\mathcal{L}(\theta)_{c=0} - \mathcal{L}(\theta)_{c=1}\right|, \tag{11}$$

where α is a hyper-parameter that controls the trade-off between fairness and accuracy.

V. Experimental Setup

Datasets: Our study evaluates the use of hardware errors to enhance fairness by analyzing disease classification based on two dermatology datasets: Fitzpatrick-17k [20] and ISIC-2019 [51], [52]. The Fitzpatrick-17k dataset contains 16,577 images across 114 skin conditions, with skin tones labeled from 1 (lightest) to 6 (darkest), categorized for analysis into light (1-3) and dark (4-6) skin groups. The ISIC-2019 dataset comprises 25,331 dermoscopic images, classified into 9 classes, where images can also be separated based on gender (male and female). Each 128x128 image undergoes data augmentation, including flipping, rotation, scaling, and autoaugment [53]. We also ensure equal sample counts across demographic groups to prevent additional bias from dataset imbalance.

Models: We obtain baseline *Pre-trained* models VGG-11 [49] and ResNet-18 [2] by training 200 epochs using the Adam optimizer on the datasets with 60% training, 20% validation, and 20% test splits. We use the *FR only* configuration as the baseline for the accuracy vs fairness trade-off. We show that the trade-off can be improved by utilizing the *F+A R* configuration. Finally, we develop our Variability-aware (VA) and Variability+Fairness-aware (VFA) re-training schemes *VA-Retrain* and *VFA-Retrain* to improve the trade-off further. The accuracy and fairness are evaluated respectively based on F1-score and EOdds/EOpp0 metrics described in Section II-A. Considering most existing works on improving fairness are software-based [54]–[56], we use *Pre-trained* and *FR*

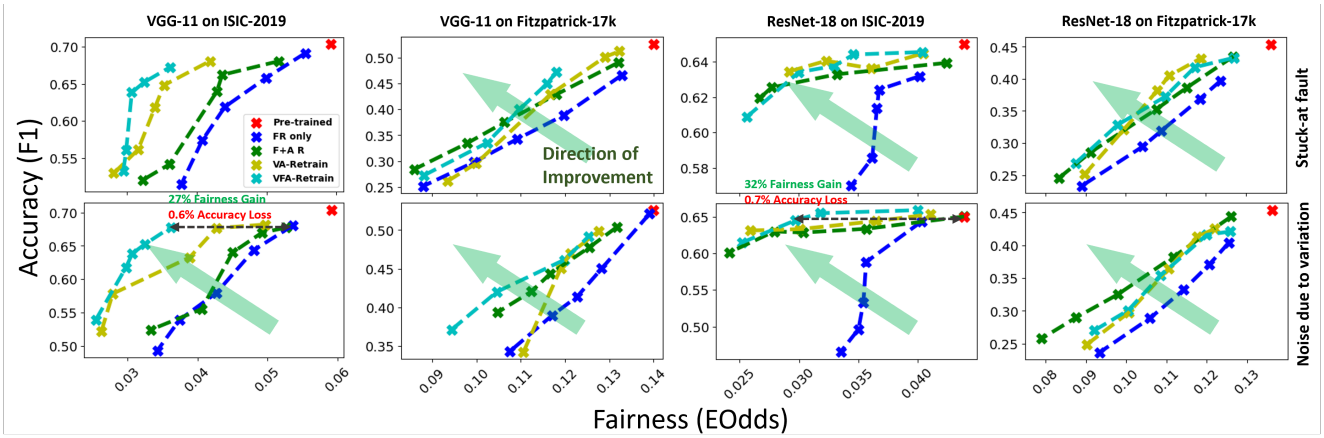

Fig. 8: Accuracy vs Fairness trade-off under simulated (a) Stuck-at-fault errors and (b) Noises due to variation. VGG-11 and ResNet-18 models have been evaluated on the Fitzpatrick-17k and ISIC-2019 datasets.

only configurations as baselines to assess our hardware-aware approach.

VI. Results

Improved Trade-off. Fig. 6 shows the accuracy and fairness metrics distribution of the VGG-11 model on simulated IMC accelerators with targeted device variability. 100 Monte Carlo (MC) simulations were performed for enough statistical significance. The "variation target" denotes the top percentile of model weights injected with errors based on ranking criteria. For instance, in (*FR only*), variation target = 5% applies errors to the top 5% of weights most important for fairness. In (*F+A R*), it targets weights in the *top* 5% of fairness ranking (most sensitive) and *bottom* 5% of accuracy ranking (least sensitive), minimizing harm to weights crucial for overall accuracy. As illustrated in Fig. 5, this approach significantly reduces selected weights for variation in most layers. As shown in Fig. 6, it improves the accuracy vs. fairness trade-off and enhances inference reliability, with higher expected accuracy for the same fairness level, as measured by EOdds.

Sensitivity to variation intensity. The accuracy and fairness of the model are also sensitive to the intensity of the applied variation for both of the error types being evaluated. As shown in Fig. 7, for the same target percentage to introduce variation, adjusting the variation intensity, represented by the stuck-at fault rate and the variability noise rate (σ'_G), also shows a trade-off between accuracy and fairness. As the amount of targeted weights is very small, this characteristic provides a valuable degree of freedom, which can be utilized to achieve better fairness in exchange for smaller accuracy degradation.

Summary of performance. Fig. 8 summarizes the averaged performance on accuracy (F1-score) vs fairness (EOdds) of the multi-objective, multi-rank weight targeting methods (FR only, F+A R) and the Variability-aware and Variability+Fairness-aware retraining methods (VA-Retrain, VFA-Retrain). The first observation is that introducing either type of device-level error in general leads to a loss of accuracy with improved fairness. Such a trade-off is manifested in the results obtained from all the methodologies evaluated. Compared to the baseline methodology that introduces errors to the subgroup of parameters based only on the ranking of fairness sensitivity (FR only), considering the ranking of both fairness and accuracy (F+A R) provides a better trade-off across all evaluated models and datasets. Our variability-aware (VA-Retrain) and variability+fairness-aware (VFA-Retrain) retraining schemes further improve the trade-off, showing a significant 32% improvement at the expense of less than 1% accuracy degradation.

VII. Conclusion and Outlook

We demonstrated how variability-induced device errors in non-ideal IMC hardware can be exploited to improve the fairness of DNN prediction. To search for an optimal way of utilizing errors to improve fairness while maintaining average accuracy, we designed a saliency-based ranking framework that strives to decouple the accuracy and fairness optimization. Our approach of weight selection informed by the multi-ranking (both accuracy and fairness) analysis and the fairness-aware retraining demonstrate a significant improvement (up to 32%) with near loss-less average accuracy. Different from the standard development of ML hardware focusing on efficiency and accuracy, our work suggests that DNN fairness could be an interesting and largely unexplored aspect of hardware-software co-design of ML hardware accelerators.

Future work could include evaluating our framework across real hardware platforms that could provide insights into the applicability of our approach. Exploring broader datasets and evaluating fairness using other metrics (such as demographic parity) may provide valuable insights into the generalizability of our framework. Research into scalable and practical error mitigation methods will further enhance the utility of our framework. Other future work directions include considering non-idealities in both crossbar arrays and peripheral circuits in IMC to develop a holistic design methodology that optimizes hardware efficiency, accuracy, and fairness. Investigating error behaviors in other ML accelerators (digital and mixed-signal) can also be investigated as an extension of our framework.

References

[1] R. Miotto *et al.*, "Deep learning for healthcare: review, opportunities and challenges," *Briefings in bioinformatics*, vol. 19, no. 6, pp. 1236–1246, 2018.

[2] K. He *et al.*, "Deep residual learning for image recognition," in *IEEE CVPR*, June 2016.

[3] Y. LeCun *et al.*, "Deep learning," *nature*, vol. 521, pp. 436–444, 2015.

[4] Y. Goldberg, "A primer on neural network models for natural language processing," *Journal of Artificial Intelligence Research*, vol. 57, 2016.

[5] H. H. Sultan *et al.*, "Multi-classification of brain tumor images using deep neural network," *IEEE access*, vol. 7, pp. 69 215–69 225, 2019.

[6] C. Qin *et al.*, "An enhanced neural network approach to person-job fit in talent recruitment," *ACM Transactions on Information Systems (TOIS)*, vol. 38, no. 2, pp. 1–33, 2020.

[7] R. Berk *et al.*, *Machine learning risk assessments in criminal justice settings*. Springer, 2019.

[8] M. Du *et al.*, "Fairness in deep learning: A computational perspective," *IEEE Intelligent Systems*, vol. 36, no. 4, pp. 25–34, 2020.

[9] Chen, Yu-Hsin and Krishna, Tushar and Emer, Joel and Sze, Vivienne, "Eyeriss: An Energy-Efficient Reconfigurable Accelerator for Deep Convolutional Neural Networks," in *IEEE ISSCC 2016*.

[10] A. Shafiee *et al.*, "Isaac: A convolutional neural network accelerator with in-situ analog arithmetic in crossbars," *ACM SIGARCH Comp. Arch. News*, 2016.

[11] N. P. Jouppi *et al.*, "In-datacenter performance analysis of a tensor processing unit," in *Proceedings of the 44th annual international symposium on computer architecture*, 2017, pp. 1–12.

[12] B. Fleischer *et al.*, "Unlocking the promise of approximate computing for on-chip ai acceleration," IBM Research Blog, 2020.

[13] I. Chakraborty *et al.*, "Resistive crossbars as approximate hardware building blocks for machine learning: Opportunities and challenges," *Proceedings of the IEEE*, 2020.

[14] "Study finds gender and skin-type bias in commercial artificial-intelligence systems," https://news.mit.edu/2018/study-finds-gender-skin-type-bias-artificial-intelligence-systems-0212 .

[15] L. H. Kamulegeya *et al.*, "Using artificial intelligence on dermatology conditions in uganda: A case for diversity in training data sets for machine learning," *bioRxiv*, 2019.

[16] K. Ferryman *et al.*, "Fairness in precision medicine. data & society. 26 feb 2018," 2022.

[17] E. Gurevich *et al.*, "Equity within ai systems: What can health leaders expect?" in *Healthcare Management Forum*, vol. 36, no. 2. SAGE Publications Sage CA: Los Angeles, CA, 2023, pp. 119–124.

[18] S. A. Ibrahim *et al.*, "Big data analytics and the struggle for equity in health care: the promise and perils," *Health equity*, 2020.

[19] M. Hardt *et al.*, "Equality of opportunity in supervised learning," *Advances in neural information processing systems*, vol. 29, 2016.

[20] M. Groh *et al.*, "Evaluating deep neural networks trained on clinical images in dermatology with the fitzpatrick 17k dataset," in *Proceedings of the IEEE/CVF CVPR*, 2021, pp. 1820–1828.

[21] A. Ankit *et al.*, "Puma: A programmable ultra-efficient memristor-based accelerator for machine learning inference," in *24th ASPLOS*, 2019.

[22] P.-Y. Chen *et al.*, "Mitigating effects of non-ideal synaptic device characteristics for on-chip learning," in *2015 IEEE/ACM ICCAD*.

[23] L. Xia *et al.*, "Stuck-at fault tolerance in rram computing systems," *IEEE JETCAS*, 2017.

[24] W. Jiang *et al.*, "Device-circuit-architecture co-exploration for computing-in-memory neural accelerators," *IEEE Transactions on Computers*, vol. 70, no. 4, pp. 595–605, 2021.

[25] Z. Yan *et al.*, "Uncertainty modeling of emerging device based computing-in-memory neural accelerators with application to neural architecture search," in *Proceedings of the 26th Asia and South Pacific Design Automation Conference*, 2021, pp. 859–864.

[26] G. Pedretti *et al.*, "Conductance variations and their impact on the precision of in-memory computing with resistive switching memory (rram)," in *2021 IEEE International Reliability Physics Symposium (IRPS)*.

[27] C. Quan *et al.*, "Training-free stuck-at fault mitigation for reram-based deep learning accelerators," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 42, no. 7, 2023.

[28] H. Shin *et al.*, "Fault-free: A framework for analysis and mitigation of stuck-at-fault on realistic reram-based dnn accelerators," *IEEE Transactions on Computers*, vol. 72, no. 7, pp. 2011–2024, 2023.

[29] B. Zhang *et al.*, "Handling stuck-at-faults in memristor crossbar arrays using matrix transformations," in *Proceedings of the 24th Asia and South Pacific Design Automation Conference*, ser. ASPDAC '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 438–443. [Online]. Available: https://doi.org/10.1145/3287624.3287707

[30] A. Antolini *et al.*, "Combined hw/sw drift and variability mitigation for pcm-based analog in-memory computing for neural network applications," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 13, no. 1, pp. 395–407, 2023.

[31] S. Lee *et al.*, "Fast and low-cost mitigation of reram variability for deep learning applications," in *2021 IEEE 39th International Conference on Computer Design (ICCD)*, 2021, pp. 269–276.

[32] S. K. Gonugondla *et al.*, "Swipe: enhancing robustness of reram crossbars for in-memory computing," in *Proceedings of the 39th ICCAD*. New York, NY, USA: Association for Computing Machinery, 2020. [Online]. Available: https://doi.org/10.1145/3400302.3415642

[33] J. Kim *et al.*, "Vcam: Variation compensation through activation matching for analog binarized neural networks," in *2019 IEEE/ACM ISLPED*.

[34] Y. Yang *et al.*, "The limits of fair medical imaging ai in real-world generalization," *Nature Medicine*, pp. 1–11, 2024.

[35] A. Agarwal *et al.*, "A reductions approach to fair classification," in *ICML*. PMLR, 2018, pp. 60–69.

[36] T. Le Quy *et al.*, "A survey on datasets for fairness-aware machine learning," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 12, no. 3, p. e1452, 2022.

[37] A. Brown *et al.*, "Detecting shortcut learning for fair medical ai using shortcut testing," *Nature communications*, vol. 14, no. 1, p. 4314, 2023.

[38] E. Raff *et al.*, "Gradient reversal against discrimination: A fair neural network learning approach," in *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 2018, pp. 189–198.

[39] Y. Ganin *et al.*, "Domain-adversarial training of neural networks," *Journal of machine learning research*, vol. 17, no. 59, pp. 1–35, 2016.

[40] M. Lin *et al.*, "Improving model fairness in image-based computer-aided diagnosis," *Nature Communications*, vol. 14, no. 1, p. 6261, 2023.

[41] I. Ktena *et al.*, "Generative models improve fairness of medical classifiers under distribution shifts," *Nature Medicine*, pp. 1–8, 2024.

[42] S. Tayebi Arasteh *et al.*, "Preserving fairness and diagnostic accuracy in private large-scale ai models for medical imaging," *Communications Medicine*, vol. 4, no. 1, p. 46, 2024.

[43] S. Hooker *et al.*, "Characterising bias in compressed models," *arXiv preprint arXiv:2010.03058*, 2020.

[44] G. Xu *et al.*, "Can model compression improve nlp fairness," *arXiv preprint arXiv:2201.08542*, 2022.

[45] Y. Wu *et al.*, "Fairprune: Achieving fairness through pruning for dermatological disease diagnosis," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2022.

[46] H. V. Pham *et al.*, "Problems and opportunities in training deep learning software systems: An analysis of variance," in *Proceedings of the 35th IEEE/ACM international conference on automated software engineering*, 2020, pp. 771–783.

[47] S. H. Nelaturu *et al.*, "On the fairness impacts of hardware selection in machine learning," *arXiv preprint arXiv:2312.03886*, 2023.

[48] Y. LeCun *et al.*, "Optimal brain damage, advances in neural information processing systems," *(NIPS 1989)*, vol. 2, 1990.

[49] K. Simonyan *et al.*, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.

[50] M. Groh *et al.*, "Towards transparency in dermatology image datasets with skin tone annotations by experts, crowds, and an algorithm," *Proceedings of the ACM on HCI*, vol. 6, no. CSCW2, pp. 1–26, 2022.

[51] M. Combalia *et al.*, "Bcn20000: Dermoscopic lesions in the wild," *arXiv preprint arXiv:1908.02288*, 2019.

[52] P. Tschandl *et al.*, "The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Scientific data*, vol. 5, no. 1, pp. 1–9, 2018.

[53] E. D. Cubuk *et al.*, "Autoaugment: Learning augmentation policies from data," *arXiv preprint arXiv:1805.09501*, 2018.

[54] E. Tzeng *et al.*, "Simultaneous deep transfer across domains and tasks," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 4068–4076.

[55] Z. Wang *et al.*, "Towards fairness in visual recognition: Effective strategies for bias mitigation," in *Proceedings of the IEEE/CVF CVPR*, 2020, pp. 8919–8928.

[56] B. H. Zhang *et al.*, "Mitigating unwanted biases with adversarial learning," in *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, 2018, pp. 335–340.