Data-Algorithm-Architecture Co-Optimization for Fair Neural Networks on Skin Lesion Dataset

Yi Sheng[1], Junhuan Yang[1], Jinyang Li[1], James Alaina[2], Xiaowei Xu[3], Yiyu Shi[4], Jingtong Hu[5], Weiwen Jiang[1], and Lei Yang[1]

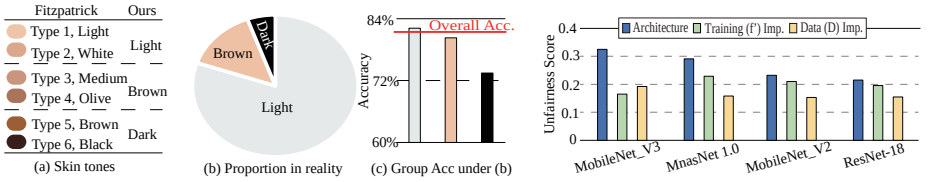
[1] George Mason University
ysheng2@gmu.edu, lyang29@gmu.edu
[2] University of Pittsburgh Medical Center
[3] Guangdong Provincial People's Hospital
[4] University of Notre Dame
[5] University of Pittsburgh

Abstract. As Artificial Intelligence (AI) increasingly integrates into our daily lives, fairness has emerged as a critical concern, particularly in medical AI, where datasets often reflect inherent biases due to social factors like the underrepresentation of marginalized communities and socioeconomic barriers to data collection. Traditional approaches to mitigating these biases have focused on data augmentation and the development of fairness-aware training algorithms. However, this paper argues that the architecture of neural networks, a core component of Machine Learning (ML), plays a crucial role in ensuring fairness. We demonstrate that addressing fairness effectively requires a holistic approach that simultaneously considers data, algorithms, and architecture. Utilizing Automated ML (AutoML) technology, specifically Neural Architecture Search (NAS), we introduce a novel framework, BiaslessNAS, designed to achieve fair outcomes in analyzing skin lesion datasets. BiaslessNAS incorporates fairness considerations at every stage of the NAS process, leading to the identification of neural networks that are not only more accurate but also significantly fairer. Our experiments show that BiaslessNAS achieves a 2.55% increase in accuracy and a 65.50% improvement in fairness compared to traditional NAS methods, underscoring the importance of integrating fairness into neural network architecture for better outcomes in medical AI applications.

Keywords: AI-powered dermatology; Fairness; Neural Architecture Search.

1 Introduction

The democratization of AI is rapidly expanding the use of machine learning, notably neural networks, across various medical disciplines [31, 33], with dermatology leading due to the availability of comprehensive skin lesion datasets [9]. However, unlike general-purpose image datasets like ImageNet [17], skin lesion datasets often exhibit biases, particularly regarding skin tone. This imbalance poses a significant challenge for machine learning in dermatology, as it can result



(a) Skin tones (b) Proportion in reality (c) Group Acc under (b)

(i) Illustration of the fairness issue in ISIC2019 dataset with different proportions in terms of skin tones

(ii) Architecture, training algorithm and data augment can jointly affect ML model fairness

Fig. 1. Bias issue behind training dataset and three fairness-related factors

in models that, while accurate on average, perform poorly for underrepresented groups. Our analysis of the ISIC2019 dermatology dataset [5] revealed a notable accuracy disparity of over 10% between lighter and darker skin tones, despite an overall accuracy of 81.71% in Fig. 1(i). This issue of skin-type bias is not unique to academic datasets but is also prevalent in commercial AI applications, including facial-analysis tools [4] and Skin Image Search platforms [16].

Researches [20, 19] have highlighted that data bias significantly impacts the fairness of machine learning (ML) models. And Fig. 1(ii) shows that except data, algorithm and network also affect the fairness, and one observation from Table 2 shows that co-optimization of these factors yields the best performance. Through a comprehensive review of the ML process, we've found that neural architectures and training algorithms, alongside data, also influence fairness. Interestingly, these factors are interconnected, suggesting that optimizing them in isolation may not yield the most equitable outcomes. While previous studies have focused on enhancing fairness from data [28, 22] or algorithmic [7, 21, 8] perspectives, the role of neural architecture remains underexplored. Neural Architecture Search (NAS), which has gained attention for improving model performance and efficiency [15, 14], involves search space formulation, architecture evaluation, and optimizer evolution. This process offers a unique avenue to integrate data processing, training algorithms, and architecture search within a unified framework, yet fairness considerations have largely been overlooked in NAS, especially regarding biomedical data.

In response, this paper introduces Biasless-NAS, a comprehensive framework that leverages NAS for the co-optimization of data, training algorithms, and neural architecture. BiaslessNAS embeds fairness awareness into each phase of the NAS process, ensuring that these elements are simultaneously optimized for fairness in skin lesion dataset analysis. This approach not only addresses the gap in incorporating fairness into NAS but also sets a new standard for developing equitable ML models in biomedical applications. Experimental results show that BiaslessNAS can achieve the highest accuracy with a fairness improvement of 33.13%. With tolerant accuracy degradation, BiaslessNAS can find a fairer neural architecture with 65.59% fairness improvements.

2 Related Work

With the biased data in hand, traditional approaches can be divided into two directions: (1) data bias removal, and (2) fair training. Data bias removal: one way to remove the bias is by building a balanced dataset, however, it is a time-consuming process. An alternative way is to employ data augmentation. For example, [6] generates biased sets to increase the minority data artificially. In addition to data balance [11, 25], techniques were proposed to modify the training algorithms in addressing the fairness issue. Authors in [24, 18] applied adversarial training and add a discrimination module to improve fairness.

Our work stands at a different point to consider the neural architecture in addressing the fairness issue. We propose a framework to jointly optimize neural architectures, training algorithms, and data augmentation. The above-mentioned debiasing methods can be integrated into our framework.

3 Method

3.1 Fairness Metric Definition and Factor Investigation

Given a neural architecture N and datasets $\langle T, D\rangle$ where T is the training set and D is the validation set, N is trained on T to generate the model f'_N, which is then validated on D to obtain accuracy $A(f'_N, D)$. Fairness exists because data in D have additional attributes (e.g., skin tones), which will divide D into groups, denoted $\{D_{g_1}, D_{g_2}, \cdots, D_{g_K}\}$. For example, if a dataset contains two skin tones (i.e., $g_1 = light\_skin$ and $g_2 = dark\_skin$), the accuracy of model f'_N on group g_i is denoted as $A(f'_N, D_{g_i})$.

We define the "unfairness score" based on the overall accuracy and the group accuracy, denoted as $U(f'_N, D)$. Specifically, in this project, we calculate the unfairness score [18] $U(f'_N, D)$ as the L1-norm:

$$U(f'_N, D) = \sum_{\forall g_i \in G} \{|A(f'_N, D_{g_i}) - A(f'_N, D)|\}. \quad (1)$$

Results in Fig 1 (ii) illustrate that different architectures (N) have different unfairness scores. We further investigate the influence of the training approach and data preprocessing. In Fig. 1 (ii), we modify the loss function in training to consider fairness in the training process, denoted as "Training Imp.", and we conduct data balancing to increase the samples in minority groups aiming at improving fairness, denoted as "Data Imp.". It is clear that both approaches can reduce the unfairness score. More interestingly, the three factors N, f', and D are coupled with each other, which indicates that optimizing them simultaneously is best to minimize the unfairness score.

3.2 BiaslessNAS Framework

Overview of BiaslessNAS framework: Fig. 2 shows the overview of BiaslessNAS, which is composed of 4 components: ① *reinforcement learning (RL)*

optimizer, ② *search space*, ③ *fairness-aware trainer*, and ④ *fairness and accuracy evaluator*. Specifically, a recurrent neural network (RNN)-based controller guides the optimization process by sampling a batch generation method (BGM) and a neural architecture (a.k.a., child network) N in the *search space*. Then, *the fairness-aware trainer* will tune the child network. Next, in the *evaluator*, the obtained model from the trainer will be evaluated to obtain accuracy and unfairness scores. With these metrics, a reward will be generated, which will be used to update RNN in the controller. We will introduce the details of these components in the following texts.

① **RL Optimizer**: The controller iteratively predicts the hyperparameters of the batch generation method BGM and the child network N. In each iteration, the controller receives a reward to update the RNN network. The reward R is generated based on the outputs of the evaluator (see ④), including accuracy $A(f'_N, D)$, and unfairness score $U(f'_N, D)$. R is computed as follows.

$$R = \begin{cases} \alpha \cdot A(f'_N, D) - \beta \cdot U(f'_N, D) & A(f'_N, D) \geq AC \\ -1 & otherwise \end{cases} \tag{2}$$

where α, β are two scaling factors that could be adjusted according to the specific demands on accuracy or fairness, and AC is the requirement of the model accuracy on the full dataset D.

Based on the reward, we employ reinforcement learning to update the controller. Specifically, we apply the Monte Carlo policy gradient algorithm [32]:

$$\nabla J(\theta) = \frac{1}{m} \sum_{k=1}^{m} \sum_{t=1}^{T} \gamma^{T-t} \nabla_\theta \log \pi_\theta(a_t | a_{(t-1):1})(R_k - b) \tag{3}$$

where m and T are the batch size and step in each episode. Rewards are discounted by an exponential factor γ, and b is the average exponential moving.

② **Data/Architecture Fusing Search Space**: The search space is composed of two sets of hyperparameters: (1) hyperparameters for BGM, and (2) hyperparameters for child network architecture N.

Batch Generation. The idea of creating BGM is to adjust the composition of data from different groups in one training data batch. We define o_i to be a ratio, indicating the percentage of images in one batch comes from sub-dataset D_{g_i}. Let BS be the batch size, then, we have $o_i \times BS$ to be the number of images from sub-dataset D_{g_i}, and we have the constraint that $\sum_{\forall g_i \in G} \{o_i\} = 1$. To avoid accuracy degradation caused by oversampling of minority groups, we additionally have the following constraint: $\forall g_i \in G, g_j \in G$, if $|D_{g_i}| \leq |D_{g_j}|$, then $o_i \leq o_j$, where $|D_{g_k}|$ indicates the size of sub-dataset D_{g_k}.

Neural Architecture. We apply a linear array of a block as the backbone architecture. The design of basic blocks is inspired by the existing popular convolutional neural networks, including VGG-Net [27], MobileNet [13], and ResNet [12]. In this work, as shown in Fig. 2 ②, we involve four types of basic blocks, including MobileNetV2-inspired ones (i.e., MB and DB), ResNet-inspired block (RB), and VGG-inspired block (CB). The basic blocks have four hyperparameters, including channel numbers ($CH1$, $CH2$, and $CH3$) and kernel sizes (K).

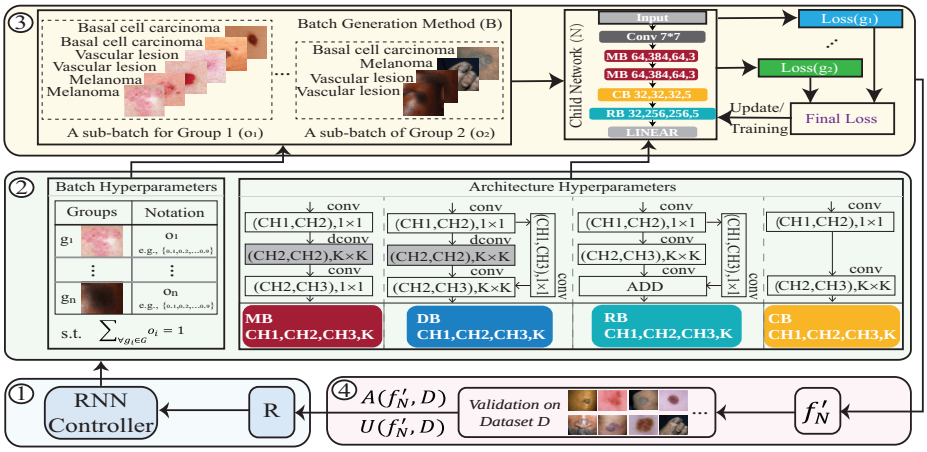


Fig. 2. Overview of BiaslessNAS : ① controller: generating a reward R and updating the recurrent neural network (RNN)-based controller; ② search space: sampling a set of hyperparameters based on the updated controller to obtain the batch composition of groups' data and a child network; ③ fairness-aware trainer: on a validated dataset, training the identified child network on the generated batches; ④ evaluator: generate the accuracy and unfairness score for the trained neural network f'_N.

Kindly note that $CH1$ is not a searchable hyperparameter. Considering two adjacent blocks ($A_i \rightarrow A_j$), $CH1$ in A_j has the same value as $CH3$ in A_i. Besides the four types of blocks, we also enable the block to be a skip operation, so that it has the flexibility in searching for the depth of the neural network.

③ **Fairness-aware Trainer**: Given an identified architecture (i.e., child network N) and BGM, the fairness-aware trainer trains the child network to generate a trained model f'_N. Specifically, we first create batches of data using BGM on the validation dataset. Then, the model is trained using a fairness-aware loss function. Finally, after the iterative training process, we can obtain f'_N.

Particularly, the fairness-aware loss function is formulated by leveraging the hyperparameters in BGM. Denote B_{g_i} as the sub-batch of samples from sub-dataset D_{g_i}, and we have $|B_{g_i}| = o_i \times |D_{g_i}|$, where $|*|$ is the size of a dataset/batch, and o_i is the ratio in BGM. For each sample $s \in B_{g_i}$, it has a target label T_s and a prediction results P_s. After the forward propagation, we apply Cross Entropy to compute the loss, as follows,

$$L = -\sum_{g_i \in G} \sum_{s \in B_{g_i}} \{\frac{\arg\max_{g_j \in G} o_j}{o_i} \cdot T_s \log P_s\}, \tag{4}$$

where $\arg\max_{g_j \in G} o_j$ identifies the ratio of the largest group to compose a batch, which is used to form a scaling factor. The final generated fair loss is also used to complete the backward propagation.

④ **Fairness and Accuracy Evaluator**: With the trained model f'_N, the accuracy can be obtained. Meanwhile, the unfairness score can be calculated based

Table 1. Accuracy (mean±standard deviation) comparisons between the existing neural architectures and BiaslessNAS using the Top-5 models trained by each neural architecture, in terms of highest reward in Eq. 2

Model	Light Acc.(%)	Dark Acc.(%)	Overall(%)	Acc Imp.	Unfair. Score	Fair. Imp.
MobilenetV2	81.90±0.78	59.26±1.2	81.69±0.77	baseline	0.2264 ±0.0194	baseline
Resnet18	82.54±1.48	63.59±1.14	82.36±1.47	0.67% ↑	0.1894 ±0.0233	16.34% ↑
ResNet34	82.95±0.69	67.18±1.14	82.81±0.67	1.12% ↑	0.1577 ±0.0181	30.34% ↑
MnasNet	76.54±1.20	61.02±3.34	76.40±1.22	5.29%↓	0.1551 ±0.0253	31.49% ↑
Biasless NAS-Fair	**79.58±0.18**	**71.79±2.57**	**79.51±0.20**	**2.18%↓**	**0.0779 ±0.0252**	**65.59%↑**
Biasless NAS-Acc	**84.37±0.53**	**69.23±1.81**	**84.24±0.52**	**2.55%↑**	**0.1514 ±0.0226**	**33.13%↑**

on the validate dataset D with Eq. 1. The obtained $A(f'_N, D)$ and $U(f'_N, D)$ will be utilized to calculate the reward in ① RL Optimizer.

4 Experiment

Dataset and settings We use the Fair and Intelligent Embedded System Challenge (ESFair) dataset [3], which is composed of data from ISIC2019, Dermnet[2], and Atlas[1]. Thera are 5 dermatology diseases for classification. We compare solutions obtained by BiaslessNAS with a set of existing neural architectures, including MobileNetV2 [23], ResNet [30], and MnasNet [29]. All models are trained from scratch with the same hyperparameters on a GPU cluster with 48 RTX 3080. The learning rate starts from 0.01 with a decay of 0.9 in 20 steps; while the batch size is 32 with 500 epochs.

Evaluation of BiaslessNAS. Table 1 reports the evaluation results. These two architectures were obtained from BiaslessNAS with the lowest unfairness score and the highest accuracy, respectively. Two hyperparameters are used in the framework: (1) Alpha is the scalable parameter for accuracy, and (2) Beta is for fairness. We explore two settings: BiaslessNAS-Fair has a larger Beta (0.8) and a smaller Alpha (0.2), while BiaslessNAS-Acc has a larger Alpha (0.8) and a smaller Beta (0.2). For a fair comparison of different neural architectures (N), all competitors are trained using the proposed fairness-aware data processing (D) and trainer (f'). As shown in Table 1, it is clear that BiaslessNAS-Fair can achieve competitive accuracy with the lowest unfairness score over others. More specifically, the unfairness score of BiaslessNAS-Fair is only 0.0779 on average, which achieves an improvement of 65.59% compared with MobileNetV2 regarding fairness. On the other hand, BiaslessNAS-Acc achieves the highest accuracy with the lowest unfairness score against other existing models.

Neural Architecture Visualization. Fig. 3(a)-(b) showcase the neural architectures derived from BiaslessNAS, highlighting the structural nuances between

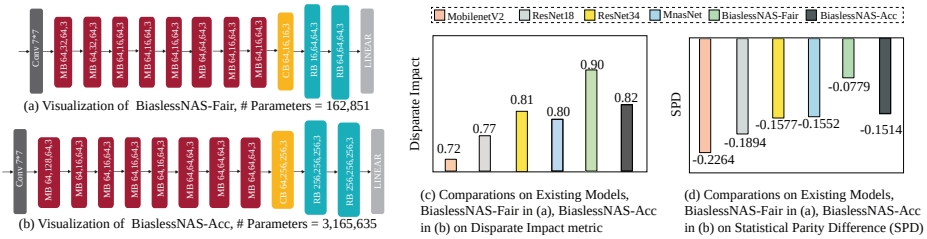


Fig. 3. Visualization of BiaslessNAS-Fair and BiaslessNAS-Acc, together with their performance on different fairness metrics

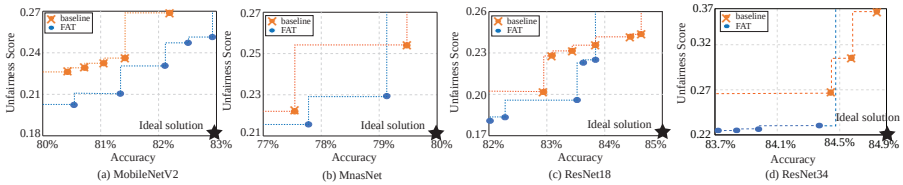


Fig. 4. Evaluation of fairness-aware trainer on the existing neural architectures

BiaslessNAS-Fair and BiaslessNAS-Acc. Despite sharing identical block types across layers, these architectures differ in the number of channels employed. Notably, both incorporate a MobileNet block at the outset for initial feature processing, followed by denser conventional and Residual blocks tailored to manage diverse group features. This visualization underscores the impact of neural architecture on fairness and suggests that strategically varying block types, particularly at the beginning and end of the architecture, can synergistically enhance fairness outcomes. This observation supports the premise that thoughtful architectural design is crucial in developing fair and effective architectures.

BiaslessNAS is Fairer on Different Metrics. In addition to the unfairness score defined in Equation 1, we further evaluate BiaslessNAS on other two commonly used fairness metrics: Disparate impact (DI) [10] and Statistical Parity Difference (SPD) [18]. Fig. 3(c)-(d) present a comparison. In Fig. 3(c), BiaslessNAS-Fair stands out by achieving the highest DI value, indicating its superiority in fairness over other examined architectures. Fig. 3(d) reveals that models with SPD scores closer to zero are preferable, with BiaslessNAS-Fair and BiaslessNAS-Acc emerging as the top performers in this regard. These findings collectively demonstrate that BiaslessNAS effectively identifies solutions that surpass conventional neural architectures in fairness across different metrics.

Evaluation of fairness-aware trainer: This ablation study is conducted by fixing the same $N\&D$ and comparing results for different f'. Fig. 4 shows the evaluation results of the fairness-aware trainer on 4 existing neural architectures. The baseline for each architecture has the setting of $\frac{o_l}{o_d} = \frac{|D_{g_l}|}{|D_{g_d}|}$, which means that the batch generator will use the same ratio between the number of dark-

Table 2. Quantitative Analysis of Three Fairness-related Factors on MobileNetV2

Models	Acc.	Unfairness	DI	Ranking
MobilenetV2 (Vanilla)	81.05%	0.2325	0.71	5
MobilenetV2 with f'	81.34%	0.2105	0.74	4
MobilenetV2 with $(D + f')$	82.14%	0.1528	0.81	2
FairNAS with N [26]	84.06%	0.1755	0.79	3
BiaslessNAS-Acc with $(D + f' + N)$	**84.24%**	**0.1514**	**0.82**	**1**

skin and light-skin images to load data. On the other hand, the fairness-aware trainer (denoted as FAT) changes the ratio of $\frac{o_l}{o_d}$ to be 1.

In these figures, each dot is associated with one solution: the dots with a cross represent the baseline approach and the dots represent the FAT approach. From the results in Fig. 4, we have several observations. (1) FAT can find neural architectures with lower unfairness scores. (2) But, if the design is to maximize accuracy regardless of the fairness, then the baseline performs better than FAT (note that one exception is MobileNetV2, in which FAT dominates the baseline approach). More specifically, when we compare the fairest architectures (i.e., the left-most dots for each approach in Fig. 4), FAT can achieve a 10.52%, 50.20%, 36.98%, and 37.82% reduction in unfairness scores on each architecture. The above results clearly show that with the same neural architecture and data augmentation, the fairness-aware trainer can indeed improve fairness but it should be careful about the possible accuracy degradation.

Evaluation of different optimization combinations. This ablation study evaluates various optimization combinations to assess the benefits of co-optimize. The results, summarized in Table 2, contrast different strategies against a baseline MobileNetV2 architecture. Initially, we examine MobileNetV2 in its standard form, followed by versions enhanced with a fairness-aware trainer (denoted as f') and then with both a co-optimized trainer and data augmentation ($D+f'$). The outcomes illustrate that co-optimization significantly enhances the fairness of MobileNetV2, as indicated by improvements in unfairness scores and disparate impact metrics. In a further analysis, a fairness-aware Neural Architecture Search (NAS), termed "FairNAS," is introduced. FairNAS seeks to identify fair neural architectures without incorporating a fairness-aware trainer or data augmentation. Interestingly, FairNAS surpasses the fairness metrics of MobileNetV2 paired with f' alone but falls short of the combination of MobileNetV2 with $D + f'$ in fairness metrics, albeit with a slight advantage in accuracy. Introducing BiaslessNAS-Acc, which integrates data-algorithm-architecture ($D + f' + N$) reveals that this approach outperforms FairNAS by achieving higher accuracy and further enhancing fairness. This comprehensive co-optimization of data, algorithm, and architecture emerges as the most effective strategy, showcasing the superior efficacy of simultaneous optimization across these dimensions for advancing both accuracy and fairness in machine learning models.

The above results give us the following three insights. (1) Neural architecture indeed affects fairness. It can even make a larger impact on fairness than the

fairness-aware trainer. (2) The neural architecture search is good at identifying architectures with high accuracy. But without the help of a fairness-aware trainer and data augmentation, it may not optimize the fairness in the search loop. (3) Co-optimization is essential to make the best accuracy-fairness tradeoff.

5 Conclusion

In this paper, we delve into the factors influencing fairness in ML systems, unveiling that optimizing models, algorithms, and data collectively can better balance accuracy and fairness. We introduce a novel framework, BiaslessNAS, designed for this holistic optimization approach, specifically targeting the inherent biases in skin lesion datasets. To ensure accuracy and fairness, BiaslessNAS incorporates a fairness-aware training mechanism that creates balanced data batches and refines weighted loss to enhance the fairness of minority groups. Additionally, a reinforcement learning optimizer steers the co-optimization process, proving that this integrated approach markedly surpasses traditional methods that optimize data, algorithms, and architecture separately. Our evaluations confirm that co-optimization significantly enhances fairness without compromising accuracy.

Acknowledgements. We gratefully acknowledge the support of the National Institutes of Health (NIH) (Award No. 1R01EB033387-01).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Dermatology atlas. http://www.atlasdermatologico.com.br/, accessed Nov, 2021
2. Dermnet dataset. http://www.dermnet.com/, accessed Nov, 2021
3. Fair and intelligent embedded system challenge at esweek 2023. https://esfair2023.github.io/ESFair/Submission.html
4. Gender and skin-type bias in commercial ai systems. https://news.mit.edu/2018/study-finds-gender-skin-type-bias-artificial-intelligence-systems-0212
5. Skin lesion analysis. https://challenge2019.isic-archive.com/
6. Abusitta, A., Aïmeur, E., Wahab, O.A.: Generative adversarial networks for mitigating biases in machine learning systems. arXiv preprint arXiv:1905.09972 (2019)
7. Bahng, H., Chun, S., Yun, S., Choo, J., Oh, S.J.: Learning de-biased representations with biased representations. In: International Conference on Machine Learning. pp. 528–539. PMLR (2020)
8. Chiu, C.H., Chung, H.W., Chen, Y.J., Shi, Y., Ho, T.Y.: Toward fairness through fair multi-exit framework for dermatological disease diagnosis. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 97–107. Springer (2023)
9. De, A., Sarda, A., Gupta, S., Das, S.: Use of artificial intelligence in dermatology. Indian journal of dermatology **65**(5), 352 (2020)
10. Feldman, M., Friedler, S.A., Moeller, J., Scheidegger, C., Venkatasubramanian, S.: Certifying and removing disparate impact. In: proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining. pp. 259–268 (2015)
11. Hao, W., El-Khamy, M., Lee, J., Zhang, J., Liang, K.J., Chen, C., Duke, L.C.: Towards fair federated learning with zero-shot data augmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3310–3319 (2021)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
13. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861 (2017)
14. Jiang, W., Yang, L., Sha, E.H.M., Zhuge, Q., Gu, S., Dasgupta, S., Shi, Y., Hu, J.: Hardware/software co-exploration of neural architectures. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **39**(12), 4805–4815 (2020)
15. Jiang, W., Zhang, X., Sha, E.H.M., Yang, L., Zhuge, Q., Shi, Y., Hu, J.: Accuracy vs. efficiency: Achieving both through fpga-implementation aware neural architecture search. In: Proceedings of the 56th Annual Design Automation Conference 2019. pp. 1–6 (2019)
16. Kamulegeya, L.H., Okello, M., Bwanika, J.M., Musinguzi, D., Lubega, W., Rusoke, D., Nassiwa, F., Börve, A.: Using artificial intelligence on dermatology conditions in uganda: A case for diversity in training data sets for machine learning. BioRxiv p. 826057 (2019)
17. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25** (2012)

18. Li, X., Cui, Z., Wu, Y., Gu, L., Harada, T.: Estimating and improving fairness with adversarial learning. arXiv preprint arXiv:2103.04243 (2021)
19. Miranda, T.C., Gimenez, P.F., Lalande, J.F., Tong, V.V.T., Wilke, P.: Debiasing android malware datasets: How can i trust your results if your dataset is biased? IEEE Transactions on Information Forensics and Security **17**, 2182–2197 (2022)
20. Nakajima, S., Chen, T.Y.: Generating biased dataset for metamorphic testing of machine learning programs. In: IFIP International Conference on Testing Software and Systems. pp. 56–64. Springer (2019)
21. Nam, J., Cha, H., Ahn, S., Lee, J., Shin, J.: Learning from failure: De-biasing classifier from biased classifier. Advances in Neural Information Processing Systems **33**, 20673–20684 (2020)
22. Ouyang, N., Huang, Q., Li, P., Cai, Y., Liu, B., Leung, H.f., Li, Q.: Suppressing biased samples for robust vqa. IEEE Transactions on Multimedia **24**, 3405–3415 (2022). https://doi.org/10.1109/TMM.2021.3097502
23. Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proc. of CVPR. pp. 4510–4520 (2018)
24. Shafahi, A., Najibi, M., Ghiasi, M.A., Xu, Z., Dickerson, J., Studer, C., Davis, L.S., Taylor, G., Goldstein, T.: Adversarial training for free! Advances in Neural Information Processing Systems **32** (2019)
25. Sharma, S., Zhang, Y., Ríos Aliaga, J.M., Bouneffouf, D., Muthusamy, V., Varshney, K.R.: Data augmentation for discrimination prevention and bias disambiguation. In: Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society. pp. 358–364 (2020)
26. Sheng, Y., Yang, J., Wu, Y., Mao, K., Shi, Y., Hu, J., Jiang, W., Yang, L.: The larger the fairer? small neural networks can achieve fairness for edge devices. In: Proceedings of the 59th ACM/IEEE Design Automation Conference. pp. 163–168 (2022)
27. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
28. Spinde, T., Krieger, D., Plank, M., Gipp, B.: Towards a reliable ground-truth for biased language detection. In: 2021 ACM/IEEE Joint Conference on Digital Libraries (JCDL). pp. 324–325. IEEE (2021)
29. Tan, M., Chen, B., Pang, R., Vasudevan, V., Sandler, M., Howard, A., Le, Q.V.: Mnasnet: Platform-aware neural architecture search for mobile (2019)
30. Targ, S., Almeida, D., Lyman, K.: Resnet in resnet: Generalizing residual architectures. arXiv preprint arXiv:1603.08029 (2016)
31. Wang, T., Xu, X., Xiong, J., Jia, Q., Yuan, H., Huang, M., Zhuang, J., Shi, Y.: Ica-unet: Ica inspired statistical unet for real-time 3d cardiac cine mri segmentation. In: International conference on medical image computing and computer-assisted intervention. pp. 447–457. Springer (2020)
32. Williams, R.J.: Simple statistical gradient-following algorithms for connectionist reinforcement learning. Machine learning **8**(3-4), 229–256 (1992)
33. Zheng, H., Han, J., Wang, H., Yang, L., Zhao, Z., Wang, C., Chen, D.Z.: Hierarchical self-supervised learning for medical image segmentation based on multi-domain data aggregation. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 622–632. Springer (2021)