Fast and Fair Medical AI on the Edge through Neural Architecture Search for Hybrid Vision Models

Changdi Yang[§1], Yi Sheng[§2], Peiyan Dong[§1], Zhenglun Kong[1], Yanyu Li[1], Pinrui Yu[1], Lei Yang[2], Xue Lin[1],Yanzhi Wang[1]
[1]*Northeastern University, Boston, MA* [2]*George Mason University, Fairfax, VA*
[1]{yang.changd, dong.pe, kong.zhe, li.yanyu, xue.lin, yanz.wang}@northeastern.edu [2]{ysheng2, lyang29}@gmu.edu

2023 IEEE/ACM International Conference on Computer Aided Design (ICCAD) | 979-8-3503-2225-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICCAD57390.2023.10323652

***Abstract*—As edge devices become readily available and indispensable, there is an urgent need for effective and efficient intelligent applications to be deployed widespread. However, fairness has always been an issue, especially in edge medical applications. Although many approaches have been proposed to mitigate the unfairness problem, their edge performance is not desirable. By examining the fairness performance of different network architectures, we observed that compared to pure convolutional neuron network (CNN) architecture, hybrid models with CNN and Vision Transformer (ViT) have exhibited better performance in terms of fairness and accuracy. After further analyzing the feature maps of intermediate layers of CNNs, ViTs, and hybrid models, we found that ViT has a strong ability to extract global information, which contributes to alleviating the unfairness problem. However, ViTs consume large amounts of computational and memory resources, which hinders their application on edge devices. To address the challenges abovementioned, we propose the first hardware-oriented co-design NAS framework to explore hybrid ViT-CNN architecture for the fair dermatology classification, namely HeViFa, which can produce light-weight models for edge devices with low unfairness scores and high classification accuracy. Experimental results show that compared with FaHaNa-Small, HeViFa-Small could search for a hybrid ViT model that reaches 10.57% and 4.03% higher accuracy as well as 0.179 and 0.0403 higher PQD score on Mix and Fitzpatrick17k dataset, repectively, and speed up by 1.21× on Samsung S21 mobile phone, 1.18× on iPhone 13 Pro and 1.37× on Raspberry Pi.**

I. Introduction

With the continuous progress of AI achieving high performance and efficiency, we have witnessed the stream of success of deep neural networks deployed on edge devices for medical applications [1], [2], e.g., mobile dermatology assistant, mobile eye cancer detection, and medical imaging and diagnostics. While many efforts have been made in the medical image analysis domain about deriving higher performance through deep learning algorithms, unfortunately, existing AI systems mainly strive for entirely high accuracy with fast on-device speed while overlooking the fairness between different human groups. The need for fairness in deep learning models was highlighted by observations stemming from a super-resolution algorithm called PLUSE [3]. Concerns were raised when it was discovered that a portrait image of Barack Hussein Obama produced a clearer output image with a white man's face after being processed by PULSE. This incident was accused of exhibiting racial bias or "racism". Furthermore, gender and skin-type biases have been identified in commercial AI systems. Reports [4] also indicate that these systems achieved an accuracy rate of 70% for the entire dataset but only 34.7% for women and a mere 17% for individuals with dark skin.

As is widely recognized, fairness presupposes that all people have the same right and should be treated equally, and there have been existing works to address fairness issues. However, current debiasing methods for fairness-aware neural networks focus on either modifying neural network models to make them interpretable on fairness [5] or by fairness-aware data collection [6]. While these approaches made important attempts at mitigating the unfairness problem, the models still do not perform well enough in terms of fairness. [7] is the most closely related work to ours, representing

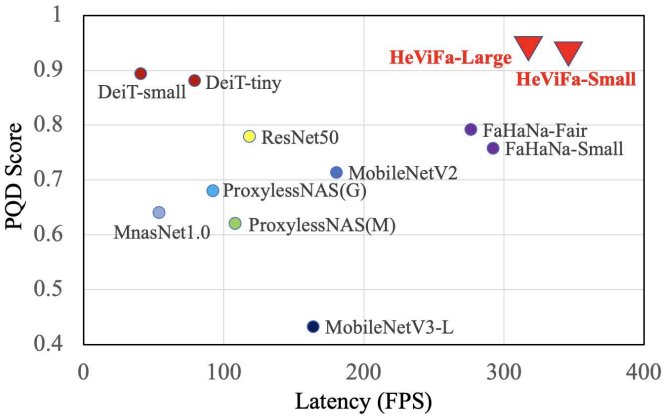


Fig. 1. Fairness comparison with current methods on Predictive Quality Disparity (PQD) vs Latency. The Latency is measured on an iPhone 13 Pro. The proposed HeViFa has a better PQD score as well as lower latency.

the state-of-art trade-offs on accuracy, speed, and fairness by a novel fairness-aware NAS framework. However, its fairness improvement is insignificant at around 1.7×. Furthermore, due to limited hardware resources on edge devices, models deployed to edge devices need to be lightweight. And [7] points out that smaller neural network models generally have lower fairness, which makes fairness an even more challenging task on edge devices. Vision Transformer (ViT) is a different architecture that has better global context extraction ability, so we analyzed ViT and found that with a similar storage size (8.8MB v.s. 8.5MB), DeiT-T has better fairness performance than MobileNet-V2 (PQD score: 0.881 v.s. 0.714). Thus, we argue that the performance of the previous attempt is limited to CNN-type only candidates. Furthermore, we attempt to figure out and explain why ViT-based models currently have superior fairness performance over pure CNN models by analyzing feature maps of intermediate layers. We discovered that ViT-based models have a larger receptive field compared with pure CNN models on minority group datasets, which leads to better performance. Nevertheless, a well-known concern is its quadratic time and memory complexity. Due to the massive number of parameters and model design, e.g., attention mechanism, ViT-based models are generally times slower than lightweight convolutional networks. Therefore, the deployment of ViT for real-time applications is particularly challenging, especially on resource-constrained hardware such as edge devices. Also, when it comes to the trade-off between task accuracy, fairness and on-device speed, it is still uncertain which choice is best among pure ViT, pure CNN, or CNN-ViT hybrid models.

Traditional methods manually fine-tune the models to achieve better fairness, but with the difficulty of quickly generating models deployable for target computing platforms. In this work, we are trying to achieve fairness through automatic neural architecture search (NAS). Although there are already some works trying to solve the fairness issue by utilizing NAS [7] and pruning [8], they are all

[§]Equal contribution

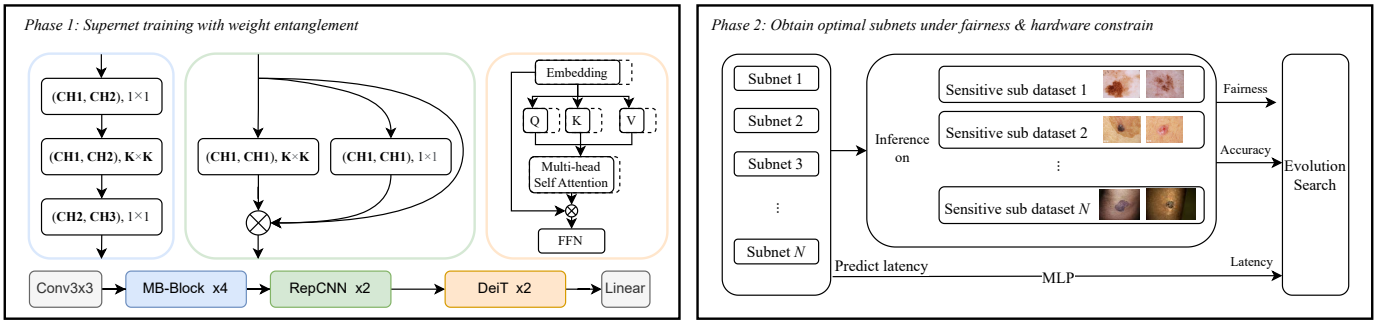


Fig. 2. Overall framework of HeViFa. In phase 1, we train the supernet with weight entanglement. The components of the supernet is shown above. After we get a well-trained supernet, we utilize the evolution algorithm to search the optimal model in phase 2. Fairness and latency constraints is integrated in the search procedure.

based on CNN, and none of them have explored the power of ViT to solve fairness problem. The fairness-aware search scheme in [7] is constrained by the convolutional structure and thus cannot be directly applied to our transformer-type layer. In addition, NAS itself is known for its lengthy search time. Therefore, a fundamental question is: Can we design and fast-generate vision models equipped with transformer layers that are more accurate, fairer, and more hardware-friendly for target devices?

In this paper, we first analyze why CNN and ViT has different performance on fairness by visualizing feature maps and find out that CNN-ViT hybrid model is superior to both pure CNN and pure ViT models. We also profile the hardware performance of different architectures and figure out that the hybrid model can reach the best trade-off between accuracy and on-device latency. Thus, we choose hybrid model as supernet in our design. To leverage the strengths of CNN-based and ViT-based operations, along with the target of deriving models that can run real-time on target edge devices, we propose a novel fairness-aware and hardware-oriented NAS framework, namely HeViFa. Given a target hardware platform, HeViFa will search for the CNN-ViT hybrid neural architectures with the best accuracy, fairness, and latency. Meanwhile, the latency can meet specific hardware specifications by our hardware-aware NAS. We also refine the NAS method with the weight entanglement strategy, which is efficient and precise for the optimization of transformer search. In addition, this framework is general for different edge computing platforms, which implies the fast deployment of AI medical applications.

This paper makes the following contributions:

- **Discovery** By visualizing intermediate feature maps, we explain why CNN and ViT models have different performances on fairness in detail and introduce a hardware-oriented convolutional operator. This paves the way for us to use CNN-based and ViT-based operators simultaneously and efficiently.
- **Framework** To the best of our knowledge, HeViFa is the first hardware-oriented NAS co-design framework to explore hybrid ViT-CNN architecture to solve unfairness problems on dermatology classification.
- **Efficiency** We can achieve the best trade-off on accuracy, fairness and latency on multiple devices, and also better NAS training efficiency by utilizing weight entanglement and latency constraints.
- **Performance** Compared with the previous state-of-the-art methods, HeViFa achieves higher accuracy and PQD (Perceptual Quality Difference) scores recorded at 82.17% and 0.937 on the Mix datasets and 80.13% and 0.87 on Fitzpatrick17k datasets with 1.18× to 1.37× speed up on multiple resource-constrained computing devices.

In the rest of the paper: Section II reviews the background and related works; Section III shows our discovery and analysis. Then we reveals our motivations. Section IV defines the problem and presents our HeViFa framework. Experimental results are shown in Section V and concluding remarks are given in Section VI.

II. Related Works

A. Fairness Mitigation Methods

Fairness has emerged as a significant concern in the machine learning (ML) community, leading to the development of various approaches aimed at addressing the issue of unfairness. Current methods for debiasing primarily concentrate on modifying datasets and optimizing training procedures.

Distribution-based methods focus on modifying the data distribution to better represent minority groups or eliminate undesired biases from the dataset. For instance, [9], [10] propose algorithms that modify objects in the dataset based on predefined rules. [11] addresses under- or over-sampling techniques to mitigate under-representation of individuals in protected groups. However, it is important to note that deep neural networks rely heavily on large amounts of data, making undersampling strategies impractical as they may reduce the data to a point where training becomes infeasible.

One-Step-Training methods are integrated into the main training procedure. Some approaches, such as [12], [13], utilize adversarial frameworks to train the model not to rely on undesired biases. However, adversarial-based methods often require annotations of protected variables or groups in the dataset, which can be a drawback due to the need for additional annotations. Additionally, optimization methods for the training process have been proposed by [8], [14]. Nevertheless, modifications to the optimization process may introduce a trade-off between fairness and accuracy, which must be carefully managed.

In contrast, we propose a vision model that directly bypasses the aforementioned drawbacks by effectively combining convolutional and self-attention operators. Our model aims to mitigate biasing and unfairness while maintaining accuracy and efficiency.

B. Vision Transformer

Transformers are initially proposed to handle the learning of long sequences in NLP tasks. Great interest has surged following the work [15] that applies a pure transformer architecture for image classification without reliance on convolutional architectures. The universality of Transformer architectures from NLP to CV is attributed to the *uniform representations* across all layers than CNNs, *self-attention mechanism* enabling early aggregation of global information, and *ViT residual connections* that strongly propagate features

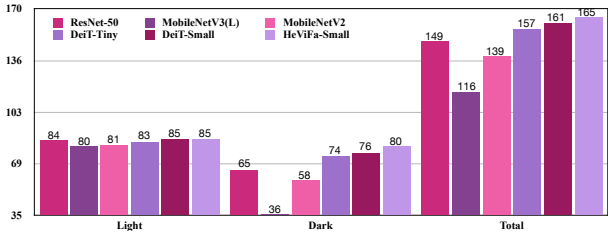


Fig. 3. Statistic of the fairness samples.

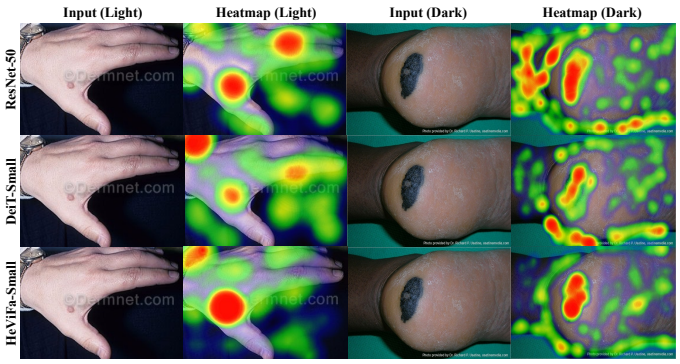


Fig. 4. Visualization of the feature map obtained by different model structures.

from lower to high layers [16]. Since then, various ViTs [17]–[21] have been proposed for different CV tasks, including object detection, semantic segmentation, and image retrieval.

Different designs of CNN-ViT hybrid models have been explored to reap the benefits of convolutions and transformers. For instance, ViT-C [22] adds an early convolutional stem to ViT. CvT [23] modifies the multi-head attention in transformers and uses depth-wise separable convolutions instead of linear projections. ConViT [24] incorporates soft convolutional inductive biases using a gated positional self-attention. Though these models can achieve competitive performance to CNNs, they still exhibit high computation and memory complexity. Unfortunately, there is a dearth of literature on hardware-oriented CNN-ViT hybrid model paradigms.

C. Neural Architecture Search

In edge AI, such as medical AI [25], fairness, accuracy, and hardware efficiency hold equal importance. The absence of any of these characteristics renders the architecture ineffective. For instance, SqueezeNet exhibits low accuracy, MobileNetV2 violates latency requirements, and MnasNet 0.5 lacks fairness [7]. Therefore, a holistic optimization approach is necessary to address all these metrics simultaneously.

Neural architecture search (NAS) methods have been developed to automatically identify neural architectures for maximum accuracy [26]. Together with the consideration of the hardware specifications, hardware-aware NAS [27], [28] further explores the hardware design space, thus jointly identifying the best architecture and hardware designs. After that, FaHaNa [7] is the first work to introduce Fairness in NAS. However, the search space of FaHaNa does not include self-attention operators, which are characterized by a large dimension definition, e.g., Q/K/V, head number, and head dimension. This potentially means an inefficient NAS method.

III. Discovery and Motivations

Through an examination of the fairness performance of various network architectures, we have observed that ViTs have demonstrated superior performance in terms of both fairness and accuracy when

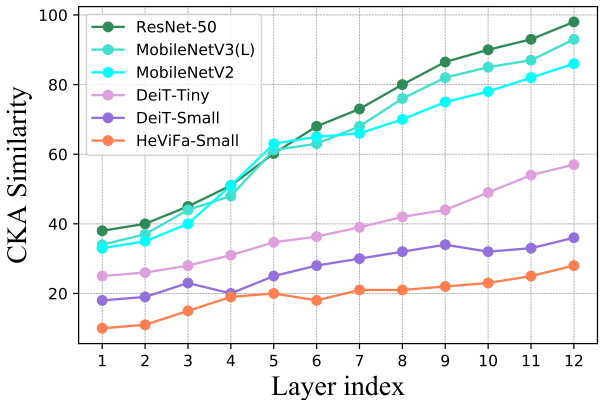


Fig. 5. CKA between each value in the last 12 output feature maps.

compared to pure CNN architectures. To optimize fairness and task accuracy, we analyze the feature maps of intermediate layers of CNNs, ViTs, and hybrid models to identify their strengths and weaknesses in visual modeling. Meanwhile, to improve network runtime speed practically, we consider on-device memory cost and degree of parallelism and introduce an efficient operator with its fusion techniques for model implementation.

Image feature modeling by different architectures. To analyze the advantages of different operators in modeling vision features, we first examine the accuracy distribution of representative CNN-based and ViT-based models, as shown in Figure 3. We conducted the analysis using a sample consisting of 100 images of the light and 100 images of the dark. The CNN-based model demonstrates significant accuracy in identifying the skin state of the light but falls short in accurately identifying the skin state of the dark. On the contrary, the ViT-based model exhibits a clear advantage in accurately identifying the skin state of the dark. According to [29], convolutional operations excel at capturing texture-level information and extracting global information by deepening the model, while self-attention operations (key operations inside ViTs) aim to directly extract abstract-level information by capturing the global receptive field. This means self-attention operations have higher robustness to the noise of background information. Next, we analyze the heatmaps of the last feature map in CNN-based (ResNet-50) and ViT-based (DeiT-Small) models for both light-skinned and dark-skinned individuals. These feature maps are crucial for the final prediction. In Figure 4, it is evident that CNN-based models precisely capture texture details in the images of light individuals, but they struggle to distinguish pathological skin conditions in the dark. This observation also indicates that the convolution operation is less robust in dealing with interference from background information. Conversely, thanks to its global analysis capability, the ViT-based model can effectively detect pathological skin conditions in the dark. Considering the distinctive characteristics of both models, our approach (HeViFa) is capable of providing a clear assessment of the skin status for both light and dark individuals. Furthermore, we calculate the centered kernel alignment (CKA) [30] in the last 12 output feature maps of the dark, which measures the average similarity between each value, as depicted in Figure 5. Our findings reveal that the CNN-based model exhibits higher similarity among different feature values compared to the other two methods. This implies that the CNN-based models are more sensitive to background noise and capture a relatively smaller amount of information than ViT-based models.

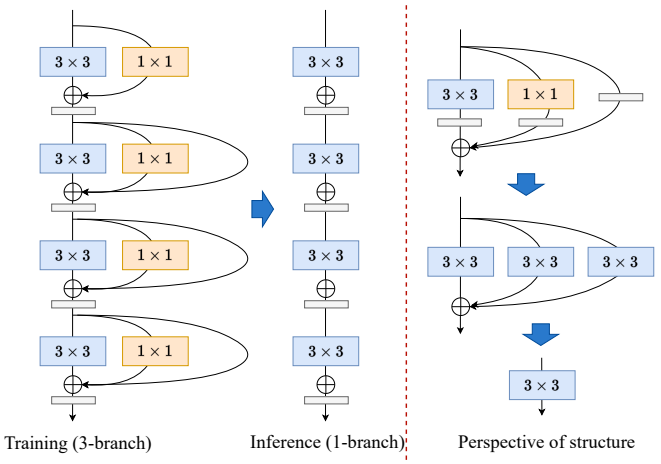


Fig. 6. **RepCNN structure**. We also show the training status and the inference status of this structure, respectively.

Fusing multiple branches into one single branch in reparameterized CNNs. The development of efficient network architectures for resource-limited devices has greatly benefited from reduced parameters and floating-point operations (FLOPs) and improved accuracy. However, conventional efficiency metrics, such as FLOPs, overlook memory cost and degree of parallelism. Multi-branch structures come with increased data movement cost, as the activation values of each branch are saved into processing engine (PE) memory or on-chip memory (if the PE memory is insufficient) to compute the subsequent tensor in the graph. Additionally, the synchronization cost arising from multiple branches impacts the overall runtime [31]. To address these challenges, we use RepCNN [32] (Figure 6) as a network component, which fuses multiple branches into more single-branch substructures during inference. This approach enables even distribution of computation among multiple PEs, preventing imbalanced computation overheads associated with multiple branches. The resulting operator fusion improves memory access and parallel computation on multiple PEs.

IV. HeViFa framework

Based on the analysis of feature modeling and network runtime overhead in Section III, we propose a fairness-aware and hardware-oriented vision model design paradigm named HeViFa , which excels at capturing features. HeViFa employs an efficient NAS algorithm to address the comprehensive optimization loop.

A. Preliminaries

Our work proposes a fair dermatology classification method with real-time inference on multiple edge devices. This section will address definitions of the problem we propose.

Fairness. In this work, we provide our definition of fairness and adapt earlier fairness metrics originally designed for binary classification or binary-sensitive attributes to our task, which involves multiple disease classes and skin types. As a result, we have derived three metrics:

(i) Predictive Quality Disparity (PQD) measures the difference in prediction quality among different skin-type groups. PQD is computed as the ratio between the lowest accuracy and the highest accuracy across the skin-type groups, as represented by the equation:

$$PQD = \frac{\min\left(\mathrm{acc}_j, j \in S\right)}{\max\left(\mathrm{acc}_j, j \in S\right)} \tag{1}$$

Here, S represents the set of skin types.

(ii) Demographic Disparity (DP) is a fairness metric that quantifies the difference in positive outcomes across different demographic or sensitive groups. It measures the percentage diversity of positive outcomes for each sensitive group, calculated as:

$$DP = \frac{1}{M}\sum_{i=1}^{M}\frac{\min\left[p(\hat{y}=1|s=j), j \in S\right]}{\max\left[p(\hat{y}=1|s=j), j \in S\right]} \tag{2}$$

In the equation, $p(\hat{y}=1|s=j)$ represents the probability of predicting a positive outcome ($\hat{y}=1$) given the sensitive attribute j (e.g., skin type, gender). S denotes the set of sensitive groups or attributes, and M represents the total number of sensitive groups.

(iii) Equality of Opportunity (EO) states that different sensitive groups should have similar true positive rates. We compute DPM and EOM across multiple skin conditions, where $m \in \{1, 2, \ldots, M\}$, using the following equations:

$$EO = \frac{\min\left[p(\hat{y}=1|y=1, s=j), j \in S\right]}{\max\left[p(\hat{y}=1|y=1, s=j), j \in S\right]} \tag{3}$$

In the equation, $p(\hat{y}=1|y=1, s=j)$ represents the probability of predicting a positive outcome ($\hat{y}=1$) given that the true label is positive ($y=1$) and the sensitive attribute is j. S denotes the set of sensitive groups or attributes.

Classification. Let D be a dataset. We define $C = \{c_1, c_2, \cdots, c_M\}$ as a set of M classes, where each data $d_i \in D$ belongs to a class $c_j \in C$. In other words, there exists a mapping function $f : f(d_i) = c_j$. A neural network N is trained to establish the mapping function from D to C. By utilizing a training dataset, N learns a function f'_N that approximates f. If $f(d_i) = f'_N(d_i)$, it signifies a correct prediction for data d_i, whereas an incorrect prediction is indicated otherwise. The accuracy $A(f'_N, D)$ represents the proportion of data in D that receives correct predictions using the model N.

Diverse Groups. In addition to the category feature (C), each data $d_i \in D$ may possess other inherent features such as skin color, race, sex, and so on. For a specific inherent feature I, it can partition the dataset D into K groups: $D = \{D_{g_1}, D_{g_2}, \cdots, D_{g_K}\}$. Let's consider skin color as an example. It can divide D into two groups: light skin (g_1 = light) and dark skin (g_2 = dark). If the number of data instances in D_{g_i} is smaller than that in D_{g_j}, i.e., $|D_{g_i}| < |D_{g_j}|$, we refer to D_{g_i} (e.g., dark skin) as the minority group compared to D_{g_j} (e.g., light skin). It is important to note that the proposed method can handle fairness considerations for more than two diverse groups.

Problem Formulation. Based on the previously defined terms, we can formally state the problem of "fairness-hardware-neural-architecture co-optimization" as follows: Given a dataset D consisting of M classes and an inherent feature I that divides D into K groups, along with a hardware configuration H and design specifications (e.g., timing constraint TC and accuracy constraint AC), our objective is to automatically generate a neural architecture N. This architecture should maximize the accuracy $A(f'_N, D)$, minimize the unfairness score $U(f'_N, D)$, and ensure that the latency $L(H, N)$ satisfies the design specifications.

In Fig. 2, we show the overall architecture of the proposed HeViFa, and it includes 2 phases. This section presents each component of HeViFa framework and its functions.

B. HeViFa framework

HeViFa Overview. In Fig. 2, we show the overall architecture of our proposed HeViFa framework. The search process is composed

TABLE I

COMPARISON WITH EXISTING WORKS. THE FIRST GROUP H.D. INDICATES HUMAN-DESIGNED MODELS AND THE SECOND FOR NAS-BASED ONES.

Group	Model	#Params	Accuracy			PQD Score	Fairness Comp.	Storage (MB)	Search Cost (GPU days)	FPS		
			Light	Dark	Average					Android	iPhone	Raspberry Pi
H.D.	DeiT-Tiny	5.9M	83.02%	73.14%	78.08%	0.881	baseline	8.80	-	15.8	79.4	0.1
	DeiT-Small	22.0M	84.49%	75.53%	80.01%	0.894	+1.30%	33.72	-	5.3	40.9	-
	MobileNetV2	2.23M	81.27%	58.02%	69.65%	0.714	-16.71%	8.51	-	83.3	180.5	0.5
	MobileNetV3(L)	4.21M	80.00%	34.57%	57.29%	0.432	-44.89%	16.05	-	78.1	164.1	0.4
	ResNet-50	23.52M	83.98%	65.43%	74.71%	0.779	-10.19%	89.72	-	30.5	118.4	1.0
NAS	ProxylessNAS(M)	2.81M	81.56%	50.62%	66.09%	0.621	-26.03%	10.70	1.8	66.8	108.2	0.2
	MnasNet1.0	3.11M	80.98%	51.85%	66.42%	0.640	-24.07%	11.86	4.4	34.4	53.9	0.3
	ProxylessNAS(G)	5.40M	83.46%	56.79%	70.13%	0.680	-20.05%	20.60	2.5	58.3	92.5	0.3
	FaHaNa-Small	4.22M	81.46%	61.73%	71.60%	0.758	-12.32%	1.61	2.1	142.9	292.3	3.0
	FaHaNa-Fair	5.50M	84.22%	66.67%	75.45%	0.792	-8.94%	20.99	2.4	134.6	276.5	1.7
	HeViFa-Small	**6.93M**	**84.82%**	**79.51%**	**82.17%**	**0.937**	**+5.64%**	**3.15**	**1.1**	**173.1**	**345.9**	**4.1**
	HeViFa-Large	**8.06M**	**85.71%**	**80.85%**	**83.28%**	**0.943**	**+6.23%**	**9.62**	**1.6**	**156.7**	**316.8**	**3.1**

TABLE II

COMPARISON WITH EXISTING WORKS ON FITZPATRICK17K. FAIRNESS COMPARISON IS CALCULATED BASED ON PQD.

Group	Model	#Params	Accuracy							PQD	DPM	EOM	Fairness Comp.
			T1	T2	T3	T4	T5	T6	Average				
H.D.	DeiT-Tiny	5.9M	70.68%	72.77%	74.17%	78.28%	82.41%	84.25%	77.09%	0.839	0.527	0.710	baseline
	DeiT-Small	22.0M	72.41%	73.57%	75.01%	78.94%	83.16%	84.73%	77.97%	0.855	0.530	0.718	+1.57%
	MobileNetV2	2.23M	56.49%	59.66%	65.68%	73.46%	83.39%	83.46%	70.36%	0.677	0.493	0.689	-16.21%
	ResNet-50	23.52M	62.48%	67.84%	72.49%	79.11%	84.99%	83.87%	75.13%	0.735	0.523	0.719	-10.38%
NAS	ProxylessNAS(G)	5.40M	59.92%	71.35%	75.15%	77.48%	83.57%	83.14%	75.10%	0.717	0.518	0.726	-12.19%
	FaHaNa-Small	4.22M	68.84%	72.73%	74.98%	76.29%	82.04%	81.74%	76.10%	0.839	0.534	0.731	+0.02%
	FaHaNa-Fair	5.50M	71.19%	74.22%	77.04%	79.35%	85.02%	84.25%	78.51%	0.837	0.522	0.734	-0.16%
	Ours-Small	**6.93M**	**75.79%**	**76.73%**	**77.80%**	**80.43%**	**85.36%**	**84.67%**	**80.13%**	**0.876**	**0.537**	**0.736**	**+4.90%**
	Ours-Large	**8.06M**	**76.41%**	**77.44%**	**78.32%**	**80.79%**	**85.81%**	**85.50%**	**80.71%**	**0.879**	**0.540**	**0.740**	**+5.15%**

of 2 phases. In Phase 1, we train the supernet with weight entanglement. After we get a well-trained supernet, we utilize the evolution algorithm to search for the optimal model in Phase 2. We utilized the weight entanglement training strategy, which is dedicated to vision transformer architecture search. The main concept is to enable weight sharing among different transformer blocks for their common parts within each layer. To elaborate further, let's consider a subnet $\alpha \in \mathcal{A}$ consisting of a stack of l layers. We represent the structure and weights of this subnet as:

$$\begin{cases} \alpha = \left(\alpha^{(1)}, \dots \alpha^{(i)}, \dots \alpha^{(l)}\right) \\ w = \left(w^{(1)}, \dots w^{(i)}, \dots w^{(l)}\right) \end{cases} \tag{4}$$

Here, $\alpha^{(i)}$ refers to the selected block in the i-th layer. Therefore, $\alpha^{(i)}$ and $w^{(i)}$ are actually chosen from a set of n block candidates within the search space, as defined by:

$$\begin{cases} \alpha^{(i)} \in \left\{b_1^{(i)}, \dots b_j^{(i)}, \dots b_n^{(i)}\right\} \\ w^{(i)} \in \left\{w_1^{(i)}, \dots w_j^{(i)}, \dots w_n^{(i)}\right\} \end{cases} \tag{5}$$

In the above formulation, $b_j^{(i)}$ represents a candidate block in the search space, and $w_j^{(i)}$ represents its associated weights. This means that for any two blocks $b_j^{(i)}$ and $b_k^{(i)}$ in the same layer, the following condition holds:

$$w_j^{(i)} \subseteq w_k^{(i)} \text{ or } w_k^{(i)} \subseteq w_j^{(i)} \tag{6}$$

The training of any block will impact the weights of others in their overlapping sections, as depicted in Figure 2.

Equipped with weight entanglement, HeViFa is capable of searching transformer architectures efficiently and effectively. Compared with classical search methods, our method could search for hybrid models quickly with lower memory costs.

Supernet Design. In Fig. 2, we show the supernet design and search space based on different basic computing blocks. Motivated by FaHaNa [7], we also utilize multi-stage architecture to extract abstract information gradually. Unlike FaHaNa using a CONV7 $\times$ 7 layer in the front, we replace it with a CONV3 $\times$ 3 layer since it will consume less memory and latency on I/O transfer. In the first two stages, we delopy CNN for its better capacity to model local information. We consider the MobileNetV2 and RepCNN blocks as candidate blocks. And search space includes channel numbers and kernel sizes. In the third stage, we take DeiT-Tiny as a candidate and the search space includes five dimensions: embedding dimension, Q-K-V dimension, number of heads, MLP ratio, and network depth. After that, a linear layer is added to perform the final classification.

Search Space. We design a search space that combines five variable factors in transformer building blocks and two variable factors in CNN building blocks. All these factors are important for model capacities.

Following one-shot NAS methods, we encode the search space into a supernet. That is, every model in the space is a part/subset of the supernet. All subnets share the weights of their common parts. The supernet is the largest model in the space, and its architecture is shown in Figure 2. In particular, the supernet stacks the maximum number of transformer blocks with the largest embedding dimension, Q-K-V dimension and MLP ratio as defined in the space. During training, all possible subnets are uniformly sampled, and the corresponding weights are updated.

Fairness constraints. After a sub-network is generated, we will evaluate it on the part of the dataset and get an unfairness score U_{eval}. In this work, we define the unfairness score as the difference in accuracy between the Light skin and Dark skin datasets. With target unfairness target U_{target}, we will get unfairness loss $L_{fair} = |U_{target} - U_{eval}|$

Latency constraints. Prior works [33], [34] either collect on-device latency data to build a lookup table for latency estimate, or deploy each candidate on chip to gather real latency data. Clearly, the existing methods mentioned above have drawbacks in terms of significant estimation errors or introducing additional overhead by relying on real on-chip latency data during the search process.

Therefore, we propose a novel approach to address these issues. Our solution involves generating a diverse set of candidate building blocks within the search space and measuring their latency on the mobile device. Subsequently, we utilize this collected data to train a deep neural network that predicts the speed or latency of candidate architectures. Remarkably, we have found that a compact DNN consisting of a few fully connected layers is sufficient for achieving this objective. An additional advantage of this approach is the once-for-all benefit, meaning that the latency model can be reused as long as the target device remains the same. Consequently, searching for new sub-networks under different constraints does not incur extra evaluation costs. The regularization term based on latency is then defined as follows: $\mathcal{L}_{reg}^{\text{LAT}} = \left|\sum_b \mathbb{S}\{o_1, i, s, s', k\} - \mathcal{S}\right|^2$

where $\mathbb{S}$ denotes the DNN to predict latency based on block characteristics (feature size, input and output channel, etc.). $\mathcal{S}$ is the target latency, and $\sum_b$ denotes latency measured by blocks.

Search Pipeline. Our search pipeline includes two phases.

Phase 1: Supernet Traning. To search ViT-based supernet with faster convergence speed and better performance, we introduce Weight Entanglement technique [35]. In each iteration, a subnet is uniformly sampled from the search space and updates its corresponding weights inside the supernet while freezing the rest.

Phase 2: Evolution Search under fairness and Latency Constraints. After Phase 1, we get a fully-trained supernet. The evolution algorithm generates a set of subnets. At the onset of the evolution search, we select N random architectures as initial seeds. From these, the top k architectures are chosen as parents for generating the subsequent generation through crossover and mutation. During each generation, two randomly chosen candidates undergo crossover to produce a new architecture, while each candidate has a probability of P_d to mutate its depth. Additionally, with a probability of P_m, each candidate mutates its blocks to create a new architecture. Following the generation of a child network, it undergoes evaluation based on fairness, accuracy, and latency. To accomplish this objective, we first input the subnet into the latency predictor to determine if it meets the hardware's latency specifications. If it fails to do so, a negative loss term is generated to further regulate the search process. In order to expedite the evaluation and enable automated optimization, we assess the performance of each block offline on the provided hardware device. This allows for efficient estimation of latency during the search process. Once the final neural network architecture is identified, we conduct an end-to-end evaluation on the target devices. The fairness and accuracy of this set of models are evaluated on a subset of the training dataset.

V. Experiments

A. Experiments Setting

1) Datasets: We use two dermatology datasets, Fitzpatrick17k and a mixed dataset. Fitzpatrick17k dataset [36] compiled 16,577 clinical images with skin condition labels and annotated them with Fitzpatrick skin-type labels. There are 114 different skin conditions, and each one has at least 53 images. They further divided these skin conditions into two more advanced categories: 3 (malignant, non-neoplastic, benign) and 9. Fitzpatrick labeling system is a six-point scale initially developed for classifying the sun reactivity of skin and adjusting clinical treatment according to skin phenotype. The Mix dataset we use is a dermatology dataset that is built on the open-access datasets, including 2019 [37]–[39] for light-skin, Dermnet [40], and Atlas dermatology [41] for dark-skin. These images are used for a classification task with five dermatology diseases: Melanoma, Melanocytic nevus, Basal cell carcinoma, Dermatofibroma, and Squamous cell carcinoma.

TABLE III
Data distribution for dermatology disease type and skin tones on Fitzpatrick17K and Mix datasets.

Dataset	Dermatology diseases	Skin tones						
		T1	T2	T3	T4	T5	T6	Total
F17K	Benign	444	671	475	367	159	44	2160
	Malignant	453	742	456	301	147	61	2160
	Non-neoplastic	2050	3395	2377	2113	1227	530	11692
	Total	2947	4808	3308	2781	1533	635	16012
Mix		Dark			Light			Total
	Melanoma	143			1533			1676
	Melanocytic nevus	111			678			789
	Basal cell carcinoma	366			1251			1617
	Dermatofibroma	147			240			387
	Squamous cell carcinoma	145			465			610
	Total	912			4167			5079

In Table III, we can observe obvious data biases occur in dermatology diseases and skin tones. In F17K datasets, the non-neoplastic group dominates the dermatology diseases with a 73% share. The T2 group has the most samples of all skin tone groups, with a proportion of 7.5 times of the minor T6 group. Similarly, in Mix dataset the Light skin tone group dominates the skin tone with 4.6 times of data than the minority Dark skin tone group. Such imbalance data distribution of sensitive groups makes unfairness mitigation a challenge.

2) Metrics: As discussed in Section IV, we use Predictive Quality Disparity (PQD), Demographic Disparity (DP) and Equality of Opportunity (EO) to measure the fairness of the models. We use accuracy to measure models' skin condition classification performance.

3) Training Settings: We start the fairness and latency-aware search from a fully pretrained supernet on the classification task. We use stochastic gradient descent (SGD) optimizer and momentum is set to 0.9, and set batch size to 64 on each GPU. For both datasets, the learning rate is set to 0.1 initially with "poly" policy and is determined as $\left(1 - \frac{\text{iter}}{\text{total\_iter}}\right)^{0.9}$ where iter refers to the current iteration number. The pretraining of supernet takes 100k iterations, while the search and fine-tune process both take 40k iterations. For the Fitzpatrick17k dataset, we carry out a three-class classification to perform an experimental task, in which the train and test sets are randomly split in an 8:2 ratio. For the Mix dataset, we perform an in-domain five-class classification using the same train-test ratio of 8:2. Images are augmented through random cropping, rotation, and flipping to boost data diversity, then resized to $224 \times 224 \times 3$. We use Adam optimizer to train the model with an initial learning rate 1×10^{-4}, which changes through a linear decay scheduler whose step size is 2, and decay factor γ is 0.9. We set the training epochs for both datasets to 200.

4) Experiment Enviroments: The Android latency is tested on the GPU of a Samsung Galaxy S21 smartphone with Qualcomm Snapdragon 888 mobile platform integrated with Qualcomm Kryo 680 Octa-core CPU and a Qualcomm Adreno 660 GPU. The compiler we deploy and test the model is TFLite. The iPhone latency is measured on the NPU of the Apple iPhone 13 Pro with an A15 processor with iOS version 16.1 on NPU. CoreMLTools is used to deploy the run-time model. The Raspberry Pi latency is tested on Raspberry 4 B with 8GB of RAM on ONNX Runtime. All results are averaged over 1,000 runs.

B. Exploration by HeViFa

In the first set of experiments, we demonstrate that HeViFa can significantly push forward the Pareto frontiers among fairness, accuracy, and model size, compared with the competitors. The efficacy of HeViFa's search engine is also evaluated.

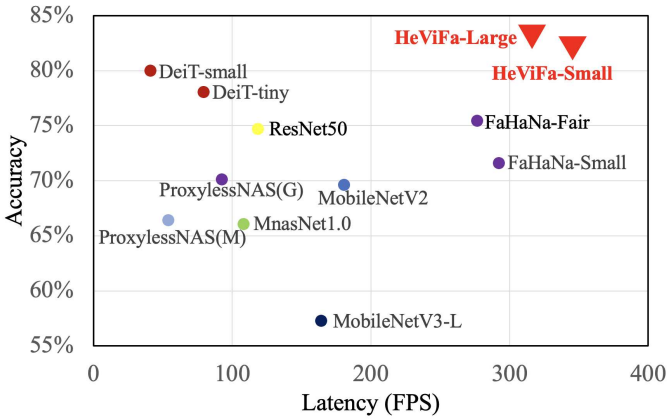


Fig. 7. Comparison with current methods on Accuracy vs Latency. The Latency is measured on an iPhone 13 Pro.

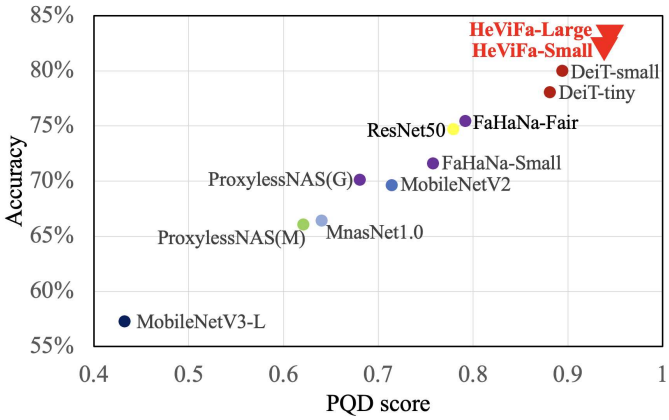


Fig. 8. Comparison with current methods on Accuracy v.s. Fairness.

1) Accuracy vs. Latency: Figure 7 reports the design space exploration results on Mix datasets, where the x-axis is the latency in Frames Per Second (FPS), and the y-axis is the classification accuracy. The ideal solution is located in the top right corner. In Figure 7, each triangle point corresponds to a HeViFa-Net and each circle is related to an existing network. From this figure, we observe that HeViFa-Small on the top right corner dominates all the existing neural networks in terms of fairness and latency; while HeViFa-Fair on the right-top corner achieves the highest fairness.

2) Fairness vs. Latency: We further investigate the trade off between fairness and latency in Figure 1. Results in Figure 1 shows the the fairness and latency performance of each candidate on Mix datasets. It consistently show that HeViFa can push forward the Pareto frontier compared with the existing neural networks. More specifically, HeViFa-Fair is the architecture that is the closest to the ideal solution. On the other hand, even HeViFa-small has the smallest size, it can still dominate most of the existing neural architectures. These two architectures will be used for further detailed comparison.

3) Fairness vs. Accuracy: We further investigate the trade off between fairness and accuracy in Figure 8. Results in Figure 8 show that with 0.937 PQD score and 82.17% of average accuracy on Mix dataset, HeViFa-Small defeat all other competitors in both metrics.

4) Search space and search cost: The efficiency and effectiveness of the weight entanglement method are evaluated by comparing GPU days. We compare the search time in Table I. There are several observations in Table I. First, thanks to the weight entanglement technique, HeViFa can significantly reduce the search space and therefore improve the search efficiency, compared with other arts. Compared with FaHaNa with 2.1 and 2.4 GPU days of search cost, HeViFa reduce it to 1.1 and 1.6 GPU days for Small and Large variations, respectively. Second, benefiting from the reduced search space, HeViFa can search for more valid architectures. Even with the larger number of parameters, the latency for multiple devices is lower and the fairness score is higher. This is because the latency and fairness constraint will pull down candidates that have slow inference speed on target device or has bad fairness performance. Overall, HeViFa can shrink the search space to examine more valid networks for better-performance architectures; meanwhile, the search time can be significantly reduced.

C. HeViFa vs. Existing Neural Architectures

Next, we compare HeViFa-Nets against competitors. We divide all neural architectures into two groups in terms of technique to produce the model. Group H.D. contains hand-designed architectures; Group NAS contains networks produced by Neural Architecture Search. We select the architecture with the best trade-off between latency and fairness from all the competitors in all groups as the baseline: DeiT-Tiny for both group H.D. and group NAS. Table III and II report the results of each competitor on the Mix dataset and Fitzpatrick17k dataset. With accuracies of 85.71% and 85.81%, and PQD scores of 0.943 and 0.879 respectively, HeViFa-Large outperformed all other competitors in the Mix dataset and Fitzpatrick17K dataset.

1) HeViFa-Small has the lowest latency: From Table I and II, we have several observations. We observed that HeViFa-Small is the fairest architecture among all competitors in both datasets. Compared with the baseline, DeiT-Tiny with a 0.881 PQD score, HeViFa-Small can get 0.937 which has a 5.64% improvement. Compared with other architectures, the fairness improvement of HeViFa-Small can reach up to 50.53% (i.e., MobileNetV3(L)). Third, HeViFa-Small has the best architecture tailored for target hardware; thus, it has the best hardware performance: 3.15M of storage, 173.1 FPS on Android phones, and 345.9 FPS on iPhone, which is far beyond real-time.

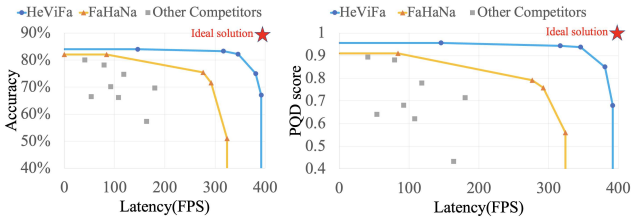
These results, in response to our initial question, verified we can find a small neural network to achieve fairness for edge devices.

2) HeViFa-Fair can achieve the highest fairness: The HeViFa-Fair model stands out as the most equitable among all competitors in both the H.D. and NAS groups, boasting a PQD score of 0.943. In comparison to the previous SOTA method, FaHaNa-Fair, HeViFa-Fair demonstrates a remarkable improvement of 0.151 on the PQD score. Even its smaller variant, HeViFa-Small, remains a formidable contender against all other competitors.

3) Pareto frontier: Figure 9 further shows the comparison of Pareto frontiers in terms of the accuracy-latency tradeoff built by all models. In Figure 9 (a), the red points form the Pareto frontier of HeViFa-Small. It is clear that HeViFa-Small dominates all other competitors. Similarly in 9 (b), HeViFa-Small dominates all architectures in fairness-latency tradeoff. These figures clearly show that HeViFa can significantly push forward the Pareto frontiers in the accuracy and model size tradeoff. All these results show the superiority of HeViFa-Nets over the existing neural architectures in terms of trade-off among latency, fairness, and accuracy.

D. Compatibility of HeViFa with Data Balancing Techiques

One typical approach for fairness improvement is to generate more minority data [6]. In table IV, we show the proposed HeViFa framework is compatible with the data balancing techniques. We apply the same method in [6] to get 5× more minority data for training. It is obvious that after data balancing, all networks except



(a) Pareto frontiers of Accuracy vs. Latency (b) Pareto frontiers of PQD score vs. Latency

Fig. 9. Pareto frontiers of Accuracy vs. Latency and PQD score vs. Latency

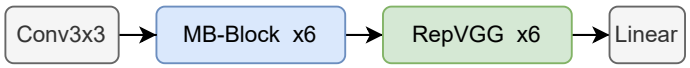
TABLE IV
RESULTS AFTER DATA BALANCING ON MIX DATASET.

Model	Accuracy			
	Light	Dark	PQD	Impr.
DeiT-Tiny	83.39%	73.45%	0.881	0.000
MobileNetV2	82.14%	66.86%	0.814	-0.080
ProxylessNAS(M)	81.53%	66.86%	0.820	0.199
MnasNet 0.5	78.82%	60.58%	0.769	0.345
MnasNet 1.0	80.20%	64.35%	0.802	0.162
FaHaNa-Small	82.02%	68.37%	0.834	0.076
HeViFa-Small	84.99%	79.74%	0.948	0.011
HeViFa-Large	85.94%	81.92%	0.953	0.010

MnasNet 1.0 can improve both accuracy and fairness; even for MnasNet 1.0, it can achieve a 0.162 higher PQD score in fairness with 0.51% accuracy degradation. From the results in Table IV, HeViFa-Large can also get benefits from data balancing to improve accuracy by 0.74% while achieving 0.010 fairness improvement. What's more, HeViFa-Large is still the fairest model.

E. Ablation Studies

In the framework we proposed, we utilized the hybrid search space of CNN and ViT as we observed hybrid models have better unfairness mitigating ability as shown in Section III. To demonstrate its advantages, we compare the performance with pure CNN and pure ViT search spaces in Table V on F17K datasets. For a fair comparison, we construct the pure CNN and pure ViT models in similar parameter numbers with the hybrid model. Detailed search space is shown in Figure 10. As shown in Table V, with similar parameter numbers, the hybrid model HeViFa-Small has 0.136 and 0.042 higher PQD score than pure CNN and pure ViT candidates.



(a) Architecture of pure CNN supernet.

(b) Architecture of pure ViT supernet.

Fig. 10. Architectures of supernet for ablation study

TABLE V
ABLATION STUDY ON THE MODEL ARCHITECTURE OF SEARCH SPACE.

Model	Parameter number	Accuracy		PQD	Fairness Comparison
		Light	Dark		
Pure CNN	6.95M	84.73%	67.84%	0.801	baseline
Pure ViT	6.90M	83.26%	74.52%	0.895	+0.094
HeViFa-Small	6.93M	85.71%	80.85%	0.937	+0.136

TABLE VI
ABLATION STUDY ON LATENCY AND FAIRNESS CONSTRAINTS.

Constraint		Accuracy		PQD	Impr.	Latency (FPS)	Impr.
Latency	Fairness	Light	Dark				
		84.83%	75.24%	0.887	baseline	257.2	baseline
✓		84.85%	75.09%	0.885	-0.002	342.5	+85.3
	✓	84.79%	79.62%	0.939	+0.052	271.7	+14.5
✓	✓	84.82%	79.51%	0.937	+0.050	345.9	+88.7

F. Insights from HeViFa

Figure 2 provides the visualization of HeViFa-Small. An insightful observation is that we applied pure CNN blocks to extract local features in the while utilizing DeiT blocks at the end layers to extract global features so as to better address the fairness issue. Such an architecture can make a good tradeoff between hardware specifications and fairness requirements: (1) thanks to re-parameterization technique and depthwise separable convolutions, the RepCNN block could speed the inference without degrading the performance. and (2) the end layers are sensitive to fairness thus ViTs are applied to achieve higher fairness. The key observation is that employing a homogeneous design with identical blocks fails to achieve a balance between accuracy, fairness, and latency. However, the HeViFa model, with its ability to flexibly select different types of blocks, overcomes this limitation and successfully achieves the desired equilibrium.

G. Experiment Results and Analysis

We showed our search results in Table I. Compared with other methods with similar resource constraints, our method reaches the highest accuracy and PQD score on both datasets . HeViFa get 0.943 and 0.879 of PQD score, respectively, much lower than other NAS methods e.g., HeViFa-Fair and ProxylessNAS(M) with the highest FPS on all devices. Furthermore, when it comes to the cost of searching, the weight entanglement technique has played a crucial role in minimizing search expenses. As a result, we have achieved the lowest search cost, requiring only 1.1 and 1.6 GPU days.

VI. CONCLUSIONS

In this work, we introduce HeViFa, an innovative hardware-oriented NAS framework aimed at addressing the issue of unfairness. It integrate fairness and hardware-aware latency in NAS to design a CNN-ViT hybrid neural architecture for the first time. By doing so, HeViFa generates a diverse set of neural architectures that significantly improve the Pareto frontier in terms of accuracy, fairness, and latency compared to existing architectures. Furthermore, HeViFa seamlessly integrates with existing techniques for enhancing fairness, making it compatible and complementary to current fairness improvement methods. HeViFa also underwent extensive experiments to assess its performance, achieving a frame rate of 173.1 FPS on a Samsung S21 mobile phone and 345.9 FPS on an iPhone 13 Pro. The accuracy and PQD (Perceptual Quality Difference) scores were recorded at 82.17% and 0.937 respectively, while on the Mix and Fitzpatrick17k datasets, it achieved scores of 80.13% and 0.876. Notably, HeViFa demonstrated superior accuracy and fairness while maintaining similar latency constraints across multiple edge devices.

ACKNOWLEDGEMENT

We gratefully acknowledge the support of National Institute Of Biomedical Imaging And Bioengineering of the National Institutes of Health R01EB033387, Army Research Office/Army Research Laboratory grant W911-NF-20-1-0167 and National Science Foundation CNS1909172 and CCF1901378.

REFERENCES

[1] O. Parraga, M. D. More, C. M. Oliveira, N. S. Gavenski, L. S. Kupssinskü, A. Medronha, L. V. Moura, G. S. Simões, and R. C. Barros, "Debiasing methods for fairer neural models in vision and language research: A survey," *arXiv preprint arXiv:2211.05617*, 2022.

[2] Z. Xu, J. Li, Q. Yao, H. Li, X. Shi, and S. K. Zhou, "A survey of fairness in medical image analysis: Concepts, algorithms, evaluations, and challenges," *arXiv preprint arXiv:2209.13177*, 2022.

[3] S. Menon, A. Damian, S. Hu, N. Ravi, and C. Rudin, "Pulse: Self-supervised photo upsampling via latent space exploration of generative models," in *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, 2020, pp. 2437–2445.

[4] L. H. Kamulegeya, M. Okello, J. M. Bwanika, D. Musinguzi, W. Lubega, D. Rusoke, F. Nassiwa, and A. Börve, "Using artificial intelligence on dermatology conditions in uganda: A case for diversity in training data sets for machine learning," *BioRxiv*, p. 826057, 2019.

[5] M. Chorać, M. Pawlicki, D. Puchalski, and R. Kozik, "Machine learning–the results are not the only thing that matters! what about security, explainability and fairness?" in *Computational Science–ICCS 2020: 20th International Conference, Amsterdam, The Netherlands, June 3–5, 2020, Proceedings, Part IV 20*. Springer, 2020, pp. 615–628.

[6] K. Choi, A. Grover, T. Singh, R. Shu, and S. Ermon, "Fair generative modeling via weak supervision," in *International Conference on Machine Learning*. PMLR, 2020, pp. 1887–1898.

[7] e. a. Sheng Yi, "The larger the fairer? small neural networks can achieve fairness for edge devices," in *the 59th DAC*, 2022, pp. 163–168.

[8] Y. Wu, D. Zeng, X. Xu, Y. Shi, and J. Hu, "Fairprune: Achieving fairness through pruning for dermatological disease diagnosis," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2022: 25th International Conference, Singapore, September 18–22, 2022, Proceedings, Part I*. Springer, 2022, pp. 743–753.

[9] E. Derman, "Dataset bias mitigation through analysis of cnn training scores," *arXiv preprint arXiv:2106.14829*, 2021.

[10] A. Stafanovičs, T. Bergmanis, and M. Pinnis, "Mitigating gender bias in machine translation with target gender annotations," *arXiv preprint arXiv:2010.06203*, 2020.

[11] S. Yan, D. Huang, and M. Soleymani, "Mitigating biases in multimodal personality assessment," in *Proceedings of the 2020 International Conference on Multimodal Interaction*, 2020, pp. 361–369.

[12] Y. Gaci, B. Benatallah, F. Casati, and K. Benabdeslem, "Iterative adversarial removal of gender bias in pretrained word embeddings," in *Proceedings of the 37th ACM/SIGAPP Symposium On Applied Computing*, 2022, pp. 829–836.

[13] D. Xu, Y. Wu, S. Yuan, L. Zhang, and X. Wu, "Achieving causal fairness through generative adversarial networks," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, 2019.

[14] S. Du, B. Hers, N. Bayasi, G. Hamarneh, and R. Garbi, "Fairdisco: Fairer ai in dermatology via disentanglement contrastive learning," in *Computer Vision–ECCV 2022 Workshops: Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part IV*. Springer, 2023, pp. 185–202.

[15] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *International Conference on Learning Representations (ICLR)*, 2021.

[16] M. Raghu, T. Unterthiner, S. Kornblith, C. Zhang, and A. Dosovitskiy, "Do vision transformers see like convolutional neural networks?" *arXiv preprint arXiv:2108.08810*, 2021.

[17] H. Bao, L. Dong, and F. Wei, "Beit: Bert pre-training of image transformers," *arXiv preprint arXiv:2106.08254*, 2021.

[18] A. Ali, H. Touvron, M. Caron, P. Bojanowski, M. Douze, A. Joulin, I. Laptev, N. Neverova, G. Synnaeve, J. Verbeek, and H. Jegou, "Xcit: Cross-covariance image transformers," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[19] K. Han, A. Xiao, E. Wu, J. Guo, C. Xu, and Y. Wang, "Transformer in transformer," in *Advances in Neural Information Processing Systems*, 2021.

[20] A. Steiner, A. Kolesnikov, X. Zhai, R. Wightman, J. Uszkoreit, and L. Beyer, "How to train your vit? data, augmentation, and regularization in vision transformers," *arXiv preprint arXiv:2106.10270*, 2021.

[21] K. Wu, H. Peng, M. Chen, J. Fu, and H. Chao, "Rethinking and improving relative position encoding for vision transformer," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 10 033–10 041.

[22] T. Xiao, M. Singh, E. Mintun, T. Darrell, P. Dollár, and R. Girshick, "Early convolutions help transformers see better," *Advances in Neural Information Processing Systems*, vol. 34, pp. 30 392–30 400, 2021.

[23] H. Wu, B. Xiao, N. Codella, M. Liu, X. Dai, L. Yuan, and L. Zhang, "Cvt: Introducing convolutions to vision transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 22–31.

[24] S. d'Ascoli, H. Touvron, M. L. Leavitt, A. S. Morcos, G. Biroli, and L. Sagun, "Convit: Improving vision transformers with soft convolutional inductive biases," in *International Conference on Machine Learning*. PMLR, 2021, pp. 2286–2296.

[25] E. Wu, K. Wu, R. Daneshjou, D. Ouyang, D. E. Ho, and J. Zou, "How medical ai devices are evaluated: limitations and recommendations from an analysis of fda approvals," *Nature Medicine*, vol. 27, no. 4, pp. 582–584, 2021.

[26] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," in *International Conference on Learning Representations (ICLR)*, 2017.

[27] H. Cai, L. Zhu, and S. Han, "Proxylessnas: Direct neural architecture search on target task and hardware," *arXiv preprint arXiv:1812.00332*, 2018.

[28] M. Tan, B. Chen, R. Pang, V. Vasudevan, M. Sandler, A. Howard, and Q. V. Le, "Mnasnet: Platform-aware neural architecture search for mobile," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 2820–2828.

[29] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[30] S. Kornblith, M. Norouzi, H. Lee, and G. Hinton, "Similarity of neural network representations revisited," in *International Conference on Machine Learning (ICML)*. PMLR, 2019, pp. 3519–3529.

[31] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[32] X. Ding, X. Zhang, N. Ma, J. Han, G. Ding, and J. Sun, "Repvgg: Making vgg-style convnets great again," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 13 733–13 742.

[33] B. Wu, X. Dai, P. Zhang, Y. Wang, F. Sun, Y. Wu, Y. Tian, P. Vajda, Y. Jia, and K. Keutzer, "Fbnet: Hardware-aware efficient convnet design via differentiable neural architecture search," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 10 734–10 742.

[34] X. Dai, P. Zhang, B. Wu, H. Yin, F. Sun, Y. Wang, M. Dukhan, Y. Hu, Y. Wu, Y. Jia *et al.*, "Chamnet: Towards efficient network design through platform-aware model adaptation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 11 398–11 407.

[35] M. Chen, H. Peng, J. Fu, and H. Ling, "Autoformer: Searching transformers for visual recognition," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 12 270–12 280.

[36] M. Groh, C. Harris, L. Soenksen, F. Lau, R. Han, A. Kim, A. Koochek, and O. Badri, "Evaluating deep neural networks trained on clinical images in dermatology with the fitzpatrick 17k dataset," 2021.

[37] P. Tschandl, C. Rosendahl, and H. Kittler, "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.

[38] N. C. Codella, D. Gutman, M. E. Celebi, B. Helba, M. A. Marchetti, S. W. Dusza, A. Kalloo, K. Liopyris, N. Mishra, H. Kittler *et al.*, "Skin lesion analysis toward melanoma detection: A challenge at the 2017 international symposium on biomedical imaging (isbi), hosted by the international skin imaging collaboration (isic)," in *2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*. IEEE, 2018, pp. 168–172.

[39] M. Combalia, N. C. Codella, V. Rotemberg, B. Helba, V. Vilaplana, O. Reiter, C. Carrera, A. Barreiro, A. C. Halpern, S. Puig *et al.*, "Bcn20000: Dermoscopic lesions in the wild," *arXiv preprint arXiv:1908.02288*, 2019.

[40] "Dermnet dataset," http://www.dermnet.com/, accessed: 2021-04-30.

[41] "Dermatology atlas," http://www.atlasdermatologico.com.br/, accessed: 2023-04-30.